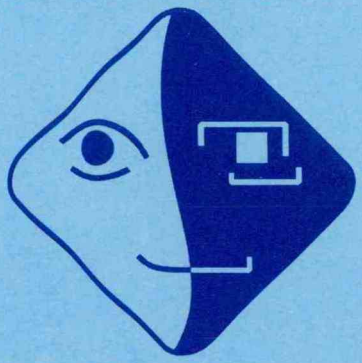
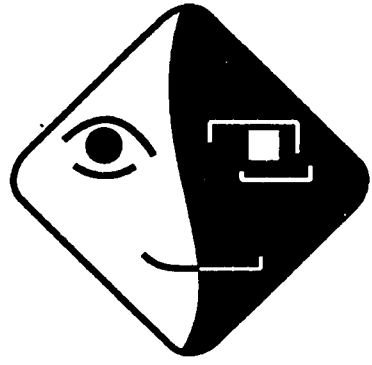
II. Magyar Számítógépes Nyelvészeti Konferencia



MSZNY 2004

Szeged, 2004. december 9-10.
http://www.inf.u-szeged.hu/mszny2004

II. Magyar Számítógépes Nyelvészeti Konferencia



MSZNY 2004

Szeged, 2004. december 9-10.
http://www.inf.u-szeged.hu/mszny2004

Kiadó: Juhász Nyomda, Szeged

Szerkesztette: Dr. Alexin Zoltán és Csendes Dóra
{alexin, dcsendes}@inf.u-szeged.hu
SZTE Informatikai Tanszékcsoport, 6720 Szeged, Árpád tér 2.
Szeged, 2004. november

Előszó

2004. december 9-10. között immáron a második Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2004) kerül megrendezésre Szegeden. A Szegedi Tudományegyetem Informatikai Tanszékcsoportja folytatni kívánja a tavaly elkezdett hagyományt, melynek keretében a számítógépes szöveg- és beszédfeldolgozás területén végzett kutatások és eredményeik ismertetésének kíván otthont teremteni. A konferencia fő célja az elvégzett vagy folyamatban lévő kutatások és fejlesztések legaktuálisabb eredményeinek bemutatása, de lehetőség nyílik hallgatói projektek, ill. a számítógépes nyelvészet ipari alkalmazásainak ismertetésére is. A felhívásokra érkezett cikkek és rövid előadások közül 46-ot tartalmaz a kiadvány.

Ezúton szeretnénk köszönetet mondani a konferencia Programbizottságának: Vámos Tibor programbizottsági elnöknek, valamint Gordos Géza, Prószéky Gábor és Váradi Tamás programbizottsági tagoknak. Szeretnénk továbbá megköszönni a Rendezőbizottság: Alexin Zoltán, Csendes Dóra és Gyimóthy Tibor munkáját.

Csirik János, a rendezőbizottság elnöke
2004. november

Tartalomjegyzék

I. Komplex nyelvi elemzések

II. Kivonatolás

III. Fordítás

IV. Tanulás, felismerés

V. Morfológia, szótár

VI. Különböző szövegtípusok elemzése

VII. Pszichológiai szempontú szövegfeldolgozás

VIII. Beszédfeldolgozás

Table of Contents

VI. Parsing different text types

VII. Psychological approach to text processing

VIII. Speech processing

I. Komplex nyelvi elemzések

A magyar INTEX fejlesztéséről

Tamás Váradi és Gábor Kata

MTA Nyelvtudományi Intézet, Budapest
{varadi,gkata}@nytud.hu

Kivonat Az INTEX rendszer egy véges állapotú technológián alapuló, integrált számítógépes nyelvelemző szoftver eszköz, melyet közel egy évtizede fejlesztett ki Max Silberztein Maurice Gross elvi irányításával. Az INTEX egyaránt kiválóan alkalmazható kutatási és oktatási célokra, és a francián kívül számos nyelvre sikerrel alkalmazták. Jelen dolgozatban beszámolunk a Nyelvtuományi Intézetben folyó munkákról, melynek célja az INTEX magyar változatának a kifejlesztése.

1. Bevezetés

A dolgozatban röviden áttekintjük az MTA Nyelvtudományi Intézetben az INTEX magyar változatának kifejlesztését célzó munkálatokat. Az INTEX rendszer egy robosztus, komplex nyelvelemző keretrendszer, amely igéretes eszköznek tűnik mind a számítógépes nyelvészeti kutató-fejlesztő munka, mind a (számítógépes) nyelvészeti oktatás számára. A rendszert francia kutatók fejlesztették ki Maurice Gross irányítása alatt, közel egy évtizedes története során számos más nyelvre (angol, olasz, szerb, portugál) is alkalmazták. A véges állapotú technológiát és az elektronikus szótárakat széles körben alkalmazó rendszer magyarra ültetése nem triviális feladat. Megoldása viszont azzal kecsegtet, hogy nemcsak saját munkánk számára nyerünk egy általánosan használható eszközt, de a nem számítógépes nyelvész szakemberek kezébe is tudunk adni egy olyan szoftvert, amellyel nyelvtant, lexikai erőforrásokat, nyelvelemző rendszereket tudnak fejleszteni.

A dolgozat az alábbi részekből áll: A 2. részben ismertetjük az INTEX főbb jellemzőit. A 3. összefoglaljuk azokat a problémákat, amelyeket a magyar változat elkészítéséhez meg kell oldani. A 4. és 5. rész bemutatja a magyar rendszerrel végzett eddigi munkát a nyílt tokenosztályok felismerése illetve a felszíni szintaktikai elemzésében.

2. Az INTEX rövid jellemzése

Az INTEX számítógépes nyelvészeti fejlesztő rendszert a Párizs VII Egyetem LADL laboratóiumában Max Silberztein készítette Silberztein (1993) és azóta is folyamatosan fejleszti. A szoftver szellemi atyja Maurice Gross, a labóratórium vezetője, akinek a lexikonra épülő, bottom-up nyelvleírási modellje Gross (1997)

egyértelműen inspirálta a rendszer elvi felépítését. A rendszer a fenti történeti okokból a francia nyelvre van leggazdagabban kidolgozva, de az angol mellett az olasz, bolgár, protugál, spanyol változatai is többé-kevésbé léteznek.[1] Magyarországon maga a fejlesztő ismertette a COMPLEX'99 conferencián, de érdemi munkálatok a magyarra csak az utóbbi két évben kezdődtek.

A rendszer meghonosítását nemcsak a robosztus és gyors véges állapotú technológia indokolja, hanem a fejlesztőknek az a kifejezett szándékuk, hogy egy viszonylag könnyen használható oktatási eszközt is adjanak a nem informatikus nyelvészek számára. Első megközelítésben az INTEX egy gyors korpuszkezelő eszköznek tűnik, amely amint betöltöttünk egy sima ascii szöveget, máris készen áll arra, hogy lekérdezhessük reguláris kifejezések segítségével. A reguláris kifejezések azonban nemcsak a szavak alakjára hanem nyelvi (morfoszintaktikai vagy akár szemantikai) jegyeikre is utalhatnak. Ezek az információk a szótári komponensből származnak, amely a rendszer központi részét képezi.

A szótár egy- illetve többtagú kifejezések tára, melyekben szóalakok találhatók, a lemmával és tetszőleges társított nyelvi információval, mindez igen hatékony véges állapotú belső reprezentációban. A rendszer egyedi sajátossága, hogy a szótár, a szöveg valamint a szövegre alkalmazott grammatika mind egyaránt véges állapotú technológiával van megvalósítva. Ami a rendszert széles körben is különösen használhatóvá teszi az a grafikai felület, amelyen viszonylag egyszerűen szerkeszthetjük és kezelhetjük a véges állapotú transzducereket (ld. példaként a 1. ábrát).

A szöveg betöltése után előfeldolgozó, normalizáló grammatikákat futtathatunk rajta, majd kiválaszthatjuk az alkalmazandó lexikai erőforrásokat. A szótárak lefuttatása azt jelenti, hogy a szöveg minden szavához társul a szótárban tárolt információ, melyeknek minden elemére külön-külön is hivatkozhatunk a további feldolgozás során. Óhatatlanul lesznek természetesen ismeretlen szavak, amelyek listáját azonnal megtekinthetjük, illetve számolnunk kell többértelmű tokenekkel. Ez utóbbiak megtekintésére igen érdekes lehetőség a szöveg mondatainak átalakítás egy véges állapotú transzducerré, melynek a kimenetét az egyes tokenek elemzése adja, és annyi bejárási utat találunk a tranzducerben, ahány féleképpen többértelmű az illető mondat. Az egyértelműsítés egyik eszköze a lexikai szűrés (a szótárak alkalmazási sorrendjének a megváltoztatásával) illetve az egyértelműsítő gráfok, azaz olyan tranzducerek, melyeknek kimenete az adott kontextusban érvényes elemzést adja.

3. A magyar változat nehézségei

A fenti vázlatos ismertetésből leszűrhető, hogy az INTEX a szótárak segítségével azonnal előállítja a szöveg morfológiai elemzését is. Ez a magyarban köztudomásulag cseppet sem triviális feladat. Különösen azért nem, mert a francia változat azt a megoldást alkalmazza, hogy az egy-egy lemmához tartozó összes képzett és ragozott alakot tételesen felsorolja a szótárban. A „felsorolást" nem kézzel kell

[1] Ld. bővebben az INTEX webhelyet: `http://www.nyu.edu/pages/linguistics/intex/`

végezni, mert szintén gráfok segítségével definiálhatjuk az egyes paradigmához tartozó alakok szerkezetét, majd a gráfok alkalmazásával előállíthatjuk az összes alakot, melyeket végül aztán bináris alakban tömöríthetünk. Akármennyire is automatizált a folyamat, a magyar morfológia gazdagsága és produktivitása nem teszi lehetővé, hogy az Értelmező Kéziszótárban szereplő összes szó valamennyi lehetséges toldalékolt alakját előállítsuk.

Az INTEX legutóbbi változata Silberztein (2004) már lehetővé teszi, hogy igazi lexikai transzducereket használjunk, azaz a szóalakok felépítését gráfokkal definiáljuk, ahol pl. a töveket tetszőleges hosszú sztringként határozzuk meg. Sajnos azonban kísérleteink Elekfi László ragozási szótárából Elekfi (1997) készített morfológiai adatbázisunk INTEX lexikai transzducerekké alakítására nem vezettek eredményre: a legenerált toldalékszekvencia halmazok implementálása meghaladta az INTEX lexikai transzducerek optimalizálási képességeit (ld. részletesebben Váradi (2005)).

Az INTEX magyar változatáról tehát akkor beszélhetünk, ha rendelkezésünkre állnak az INTEX szótárak olyan változatai, amelyek kellő számú lemmával rendelkeznek (ideálisan legalább az ÉKSz. mintegy hetvenezernyi címszavával), tartalmazzák a morfológiai jellemzőket és alkalmasak az egyes címszavakhoz tartozó valamennyi szóalak felismerésére. Ebben az esetben válik ugyanis az INTEX tetszőleges magyar szöveg számára is használhatóvá.

Természetesen addig is, amíg a jelenleg folyó intenzív munkálatok végleges eredményre nem vezetnek, az a lehetőség már most is adott, hogy meghatározott korpusz számára, az abban előforduló valamennyi szóalakot elemezzünk külső eszközzel, például a HUMOR-ral Prószéky és Tihanyi (1996), majd az elemzés kimenetét INTEX DELAF szótárfile-okba építsük be, és ezeket kompiláljuk. Nyilvánvaló, hogy ez csak korlátozott megoldás, hiszen csak arra a korpuszra teszi alkalmassá az INTEX használatát, melynek szókincséből a szótárat előzetesen összeállítottuk. Addot feladatok elvégzéséhez azonban ez a testreszabott megoldás is célszerűnek bizonyult: az INTEX-et sikeresen használtuk eddig is különböző nyelvtechnológiai projektekben. A továbbiakban az ilyen, egyedileg kipreparált szótárakra épülő munkálatokról számolunk be. (A magyar morfológia INTEX-ben történő implementálásáról ld. még Nagy Viktor és Vajda Péter tanulmányát a jelen kötetben.)

4. Tulajdonnévfelismerés az INTEX segítségével

A munka első fázisaként az alábbi tulajdonnév-típusokkal foglalkoztunk : személynevek, helységnevek, illetve intézmények nevei. [2] Abból az alapfeltevésből indultunk ki, hogy ezek a - többnyire több szóból álló - kifejezések általában tartalmaznak egy vagy több, nyílt tokenosztályhoz tartozó szót, valamint legalább egy úgynevezett *horgonyt*, vagyis olyan, zárt halmazhoz tartozó szót, amely segít a

[2] A tulajdonnévfelismerő rendszer Intexes változatának kidolgozásakor Sass Bálint korábii munkáját vettük alapul. Az adatok gyűjtését és a reguláris kifejezések szerkesztését elsősorban Sass Bálint végezte Chinchor által definiált alapelvek Chinchor et al. (1999) felhasználásával.

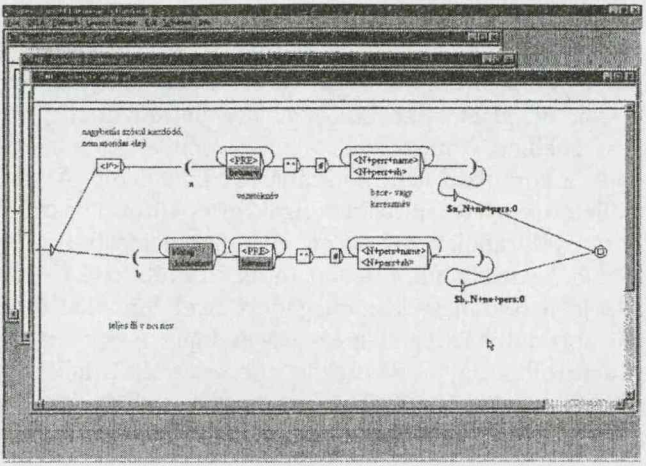
tulajdonnév típusának meghatározásában. Ilyen horgony például a személyneveknél a *Dr.* vagy *Ifj.* előtag, vagy a helységneveknél az *utca*, *tér* utótag. Az általunk alkalmazott módszer a tulajdonneves szerkezetek e két összetevőjének kezelésére épül. A tulajdonneves szerkezetek felismeréséhez és kategorizálásához az Intex szótárait, illetve szótári lookup-programját, valamint szó-szintű és szintaktikai transzducereket használtunk. A szövegelemzéshez többféle szótár áll rendelkezésünkre: az egyszerű szavakat tartalmazó DELAF szótár, a többszavas kifejezéseket tartalmazó DELACF szótár, valamint mindkét típusból a reguláris kifejezéseket is tartalmazó, nyílt tokenosztályok leírásához használt szótár-transzducerek. A tulajdonnévfelismeréshez valamennyi típusú szótárra szükségünk volt, valamint felhasználtunk egy kézzel összeállított, többszavas szótárban kódolt kész tulajdonnév-listát, melyet a munkánk során ma is folyamatosan bővítünk.

A különböző típusú tulajdonnevek részét képező, zárt halmazhoz tartozó horgonyokról listákat készítettünk, majd ezeket a horgonyokat a Delaf (egyszavas) és a Delacf ("compound words", többszavas) szótárakban szemantikai jegyekkel láttuk el, melyek a legtöbb esetben egyértelműen azonosítják annak a tulajdonneves szerkezetnek a típusát, melyben az adott horgony előfordulhat. [3] Ezzel párhuzamosan kezeljük azokat az egyszavas tulajdonneveket, melyek nyílt tokenosztályhoz tartoznak (ilyen például a -né végű asszonynév). Ezek felismeréséhez olyan lexikai transzducereket alkalmazunk, melyek szó-szintű reguláris kifejezésekkel írják le a tulajdonnév szerkezetét. A horgonyok felismerése után a következő lépésben reguláris nyelvtanokkal meg kell találnunk azokat a szavakat, melyek a horgony körül még a tulajdonneves szerkezethez tartoznak. Az Intex erre olyan eszközt kínál, mely a szintaktikai elemzést végző transzducerekhez hasonlóan működik, ám a kimenete - a szótári lookup programhoz hasonlóan - nem közvetlenül a szövegfájlba, hanem a szöveg szókincsét tartalmazó szótár-fájlok egyikébe kerül. A nyelvtani elemzés későbbi lépései számára ez azért előnyös, mert így a felismert tulajdonneves szerkezet egy szótári egységként kezelhetjük. További előnye ennek a megoldásnak, hogy mivel a tulajdonnév-nyelvtanokkal felismert sztringeket egy fájlban tároljuk, a fájl kézi ellenőrzése után a helyesnek talált tulajdonnevek hozzáadhatók a Delacf szótárhoz, melyek a kézzel előállított tulajdonnévlistát tartalmazzák. A folyamat utolsó lépéseként az előállt tulajdonnévlista alapján annotáljuk a szöveget, vagyis a felismert sztringeket főnévi csoportként címkézzük, megjelölve tulajdonnév-státuszukat és az alkategóriát, amelyhez tartoznak.

5. Szintaktikai elemzés

Munkánk célja egy olyan nyelvtani eszközlánc létrehozása az Intex korpuszfeldolgozó rendszerben, mely képes magyar mondatokból álló szövegek nyelvtani elemzésére, a tokenizálástól a mondatszegmentáláson, morfológiai elemzésen, egyértelműsítésen és a tulajdonnévfelismerésen át egészen a részleges szintaktikai elemzésig. A következőkben a szintaktikai elemzés menetét fogom bemutatni.

[3] Ez azonban nem mindig van így, hiszen például a személynevek szerepelhetnek közterületek vagy intézmények nevében is.



1. ábra. INTEX gráf a személynevek felismerésére

Részleges szintaktikai elemzés alatt egymásra épülő lokális nyelvtanokkal megvalósított elemzést értek, melynek nem az a célja, hogy a mondat teljes elemzési fáját felépítse, hanem bizonyos, lokálisan megragadható nyelvtani szerkezetek és ezek közötti dependencia-viszonyok feltérképezésére íródott. A nyelvtan képes feltárni az alapvető szintaktikai viszonyokat akkor is, ha a bemenetnek van olyan része, amit nem ismer fel. A mondat központi elemének az állítmányt képviselő finit igét tekintem, az összetevők dependencia-viszonyai alatt pedig elsősorban az igei vonzatkeretbe tartozó elemeknek és az ige szabad határozóinak felismerését értem. A módszer arra az előfeltevésre épül, hogy az igei állítmány teljesen felépített frázisokkal - illetve ezek fejével - lép régens-argumentum viszonyba. A fent megfogalmazott feladat eszerint az alábbi részfeladatokra osztható:

1. a frázisok megtalálása és címkézése
2. a tagmondathatár megtalálása
3. az igei argumentumszerkezet azonosítása
4. az igéhez tartozó szabad határozók, tagadószók, illetve a mondathatározók megtalálása

(A fenti sorrend megfelel a nyelvtanok alkalmazási sorrendjének, kivéve a 4)-t, melynek egyes elemei (igekötő, tagadás felismerése) a 2) nyelvtanok után következnek.)

Az 1-4) feladatokat végző nyelvtani szabályok egyaránt használják egymás kimenetét és az alsóbb szintű nyelvi elemzések információit. A feladatokat Intex gráfokkal ábrázolt reguláris nyelvtanok látják el, melyek kimenete a szövegbe illesztett annotáció - erre hivatkoznak az elemzés későbbi lépései. Ez a módszer bizonyos szempontból ugyan rugalmatlanná teszi a nyelvtant (megnehezíti a szabályok sorrendjének módosítását), ám ezt a hátrányt ellensúlyozza az az

előny, hogy szabályaink a szintaktikai elemzés minden pontján hivatkozhatnak bármilyen típusú információra. Így kisebb annak az esélye, hogy a mondattani elemzés egy alacsonyabb szintjén fel nem ismert sztring az egész mondatot elemezhetetlenné teszi, mivel egy későbbi lokális nyelvtannak még így is helyes környezeteként szerepelhet. A megközelítés másik előnye, hogy bármelyik lépésben módosíthatjuk a korábban lefutott szabályok kimenetét. A szabályokat az Intex grafikus felületét (a nyelvtanírásra szolgáló gráf-editort) használva készítettük és teszteltük. A nyelvtanok futtatása azonban parancssorból is végezhető, így egy olyan batch-fájlt készítettünk, mely az Intex mintaillesztő és szótári lookup programjait futtatja a paraméterként megadott nyelvtanokkal és szótárakkal, általunk definiált sorrendben. Az elemzés végeredménye egy txt fájl, mely az összes annotációt tartalmazza, de a hatékonyabb tesztelés érdekében az elemzés minden lépésének kimenetét külön fájlban tároljuk. A szabályok megalkotásához és teszteléséhez egy MTI üzleti rövidhírekből álló, 231 ezer szavas korpuszt használtuk.

5.1. A szintaktikai elemzés menete

A fent leírt négy részfeladatból jelenleg az 1-2) -re rendelkezünk részletesen kidolgozott nyelvtanokkal, így a továbbiakban ezeket mutatjuk be. Az 1) feladat, azaz a kötött szórendű frázisok megtalálása és címkézése magában foglalja a főnévi, melléknévi és a névutós csoportok azonosítását. Ezek közül a főnévi csoportnak nemcsak a határait, hanem a fejét, valamint annak lényeges morfoszintaktikai és szemantikai tulajdonságait is megadjuk.

Az 1) feladatot öt különböző szintaktikai gráf végzi. Ezek közül a legfontosabb, az NP felismerését végző nyelvtan alapját az osztályon az NKFP 2/017/2001 projektum keretei közt készült, főnévi csoportok annotálására szolgáló nyelvtan kibővített, tesztelt változata szolgáltatja Váradi (2003). Az első NP nyelvtan az összes olyan főnévi csoport lefedésére készült, melyekben a főnévi fej nincs koordinálva. Természetesen a főnévi csoportot leíró gráf tartalmaz önrekurziót, hiszen a főnévi fej (melléknévi, igenévi, birtokosi) bővítményei bővíthetők további főnévi csoportokkal. Ezt egy beágyazott algráffal oldottuk meg, mely az eredeti NP-gráffal mindenben megegyezik, ám nincs saját kimenete - így a beágyazott NP-k nem kapnak saját annotációt. A következő nyelvtannak az a feladata, hogy az előző lépésben felismert főnévi csoportok fejét megjelölje és ellássa azokkal a szintaktikai és szemantikai jegyekkel, melyek az elemzés során később relevánsak lehetnek. Ezeket a jegyeket a szótár tartalmazza a megfelelő főnévi lemmához társítva (kivéve a főnév esetét kódoló jegyet, mely a morfológiai elemzés kimenete, de szintén a Delaf szótárba kerül). Az NP esetének jelölését a következő, a mellérendelt NP-k felismeréséért felelős szabály is használja, hiszen csak azonos esetben álló főnévi csoportok koordinálhatók. A koordinációs szabály egyszerűen felülírja az első NP-szabály kimenetét: ahol két, mellérendelő kötőszóval összekapcsolt főnévi csoportot talál, melyek azonos esetben állnak, ott a köztük lévő frázishatárokat eltörli. Az 1) feladat magában foglalja a főnévi csoporton kívül a mellékneves és névutós frázisok megtalálását éas címkézését

is. Ezek a kötött szórendű frázisok alkotják a szöveg első szintű szintaktikai annotációját, és egyben a tagmondathatár-kijelölő nyelvtanok bemenetét.

Nyelvtanunk elsősorban az ige által vezérelt dependencia-viszonyok, azaz az igei vonzatok és szabad határozók feltérképezésére készült. Ez azzal a következménnyel jár, hogy a finit ige nélküli (tag)mondatokban nem azonosítunk állítmányi szerepű összetevőt. Állítmánynak mindig a finit igét tekintjük, és az állítmány kijelölése megelőzi a tagmondathatárok kijelölését. Az a feltevésünk, hogy bár nem minden tagmondat tartalmaz finit igét, a finit igék mind külön tagmondathoz tartoznak . Így a tagmondathatár megtalálását célzó szabályok hatékonyságát azon mérhetjük, hogy sikerül-e minden két, azonos mondatban lévő finit ige között határt találnunk. A tagmondathatárra azért van szükségünk, mert feltételezzük, hogy azok az összetevők, melyek az igével dependencia-viszonyban állnak, azonos tagmondatban vannak vele (kivéve a tagmondat kategóriájú vonzatokat és bővítményeket).

Mivel a tagmondat szerkezetét a szabad szórend és a gyakori szerkezeti homonímia miatt rendkívül nehéz lenne leírni, ezért nem vállalkoztunk arra, hogy olyan reguláris nyelvtant írjunk, amely bemenetének két végén a két finit ige áll, és a köztük lévő összes pontot megvizsgálva dönti el, hol lehet köztük a határ. Ehelyett egy hatlépcsős lokális nyelvtant építettünk, mely azokat a horgonyokat (kötőszók, egyes névmások, központozás) veszi alapul, melyek közül - feltételezésünk szerint - valamelyik minden határon jelen van. Ehhez megvizsgáltuk a kötőszók, kérdőszók, vonatkozó névmások és a központozási jelek, azaz a lehetséges "horgonyok" disztribúcióját , és olyan nyelvtant építettünk, mely a horgonyok jelenlétén kívül a szöveg morfológiai és alapszintű szintaktikai elemzésére támaszkodik, azaz figyelembe veszi például a frázishatárokat. Így a már felismert frázisokon belülre nem kerülhet tagmondathatár. A hatlépcsős nyelvtant kiegészítettük egy guesser-szerű szabállyal. Erre azért van szükség, hogy ellenőrizzük az előző szabályok működését. Azokat az eseteket kell újra megvizsgálni, amikor a fentiek alapján feltételezzük, hogy a két ige között van határ, ám a korábbi szabályokkal nem találtuk meg: ilyenkor a kérdéses szövegrészben minden kötőszót, kérdőszót, illetve tagmondatok elválasztására alkalmas központozási jelet megjelölünk lehetséges határként.

Az alábbi példa illusztrálja az elemzés kimenetét az 1-2) feladatok végrehajtása után:

[clauseboundary] **[np** A fogyasztói árak **<head=árak case=nom>** **np]** **[np** júniusban **<head=júniusban case=ine>** **np]** **[np** 0,3 százalékkal **<head=százalékkal case=ins>** **np]** **[predicate** nőttek predicate] **[Postp** **[np** májushoz **<head=májushoz case=all>** **np]** képest **Postp]**, **[clauseboundary]** és **[np** 10,5 százalékkal **<head=százalékkal case=ins>** **np]** **[predicate** haladták meg **predicate]** **[np** a 2000 júniusi szintet **<head=szintet case=acc>** **np]**, **[clauseboundary?]** **[np** az első fél évben **<head=évben case=ine>** **np]** **[np** az árak **<head=árak case=nom>** **np]** átlagosan **[np** 10,4 százalékkal **<head=százalékkal case=ins>** **np]** **[predicate** voltak **predicate]** **[AdjP** magasabbak **AdjP]**, **[clauseboundary]** mint **[np** egy évvel **<head=évvel case=ins>** **np]** korábban **[clau-**

seboundary] - [predicate közölte predicate] [np a Központi Statisztikai Hivatal <head=Hivatal case=nom> np] [np szerdán <head=szerdán case=sup> np].

6. Összegzés és további teendők

A dolgozatban igen vázlatosan áttekintettük az INTEX rendszert, amely igen alkalmas szoftvereszköznek ígérkezik kutatási és oktatási célokra egyaránt. Magyarra alkalmazásának elsőszámú feltétele a nyitott szótári rendszer megalkotása, ami egyben a szövegek azonnali morfológiai elemzését is biztosítja. E cél érdekében jelenleg is intenzív munka folyik együttműködésben Max Silberzteinnel, a rendszer fejlesztőjével. Időközben készül az INTEX teljesen átdolgozott változata, a NOOJ, amely várhatóan technikailag is megfelel a magyar morfológia kihívásainak.

Hivatkozások

Chinchor, N., Brown, E., Ferro, L. és P, Robinson. Named Entity Recognition Task Definition. Technical report, MITRE, 1999.

Elekfi, László. *Magyar Ragozási Szótár.* MTA Nyelvtudományi Intézet, 1997.

Gross, Maurice. The Construction of Local Grammars. In: Roche, E. és Schabes, Y. szerk. *Fininte State Language Processing.* The MIT Press, Cambridge, Mass., 1997, 329–352.

Prószéky, Gábor és Tihanyi, László. HUMOR – A Morphological System for Corpus Analysis. In: *Proceedings of the first TELRI Seminar in Tihany,* Budapest. 1996, 149–158.

Silberztein, M. *Dictionnaires électroniques et analyse automatique de textes: Le systeme Intex.* Masson, Paris, 1993.

Silberztein, Max. *INTEX Manual.* Université de Franche-Comté, http://www.nyu.edu/pages/linguistics/intex/, 2004. Translated by Michael Long, Université de Moncton.

Váradi, T. Főnévi csoportok annotálásaCLaRK rendszerben. *In: Alexin Zoltán - Csendes Dóra (szerk.): A Magyar Számítógépes Nyelvészeti Konferencia 2003 rendezvényen elhangzott előadások kötete, Szegedi Tudományegyetem Nyomdája,* 2003, 65–71.

Váradi, Tamás. On Developing the Hungarian INTEX system. In: *Proceedings of the 7th INTEX conference,* Tours. 2005.

LiLe projekt: Adatbázis mint "dinamikus korpusz"

Bódis Zoltán, Kleiber Judit, Szilágyi Éva, Viszket Anita

Pécsi Tudományegyetem Bölcsészettudományi Kar Nyelvtudományi Tanszék
7624 Pécs, Ifjúság útja 6., lile@btk.pte.hu

Abstract. A LiLe-projekt fő célkitűzése egy nyelvészeti lexikon MS-SQL-adatbázis-formában való felépítése. A kidolgozott adatbázis-struktúránk legfontosabb tulajdonsága, hogy a grammatika különböző szintjein létező szabályokat a morfémákkal azonos módon tárolja, így valósítva meg a legteljesebb lexikalizmust. A projektünk a grammatikus szóalakok és mondatok elemzésére és generálására vállalkozik. Ezeknek az eszközöknek a segítségével hozunk létre egy olyan korpuszt, amely az összes lehetséges szóalak és mondat generálására alkalmas. A generálási algoritmus kiválasztott adatbázisbeli elemeken (morfémákon és szabályokon) hajtható végre, így kívánjuk támogatni mind a nyelvészeti kutatásokat, mind a magyar nyelv oktatását. Ezt a speciális célokat megvalósító és sajátos felépítésű nyelvészeti lexikont nevezzük dinamikus korpusznak.

1 Célkitűzéseink

Kutatócsoportunk egy nyelvészeti adatbázis kifejlesztésére és feltöltésének koordinálására vállalkozott. A készülő adatbázis egy nyelvészeti lexikon, sajátos tulajdonságokkal. Szabad és kötött morfémák (tövek és toldalékok) egyaránt elemei a szótárnak, ezekből és a szintén adatbázisban tárolt szabályokból lehet (toldalékolt) szavakat előállítani. Mivel az alapok kidolgozásánál egy erősen lexikalista elméletre (GASG [1] [2]) támaszkodtunk, mindent a lexikai egységek leírásába kódoltunk bele, ami így egyszerre hordoz információt minden nyelvi szintről (fonológia, morfológia, szintaxis, szemantika).

Legfontosabb célunk egy leíró nyelvtan megalkotása, amiben (összhangban a lexikalista szemlélettel) nem a hagyományos utat követjük szabályok és kivételek felvételével. Ehelyett az egyedi eseteket rögzítjük – az egyes lexikai egységek tu-lajdonságait, viselkedését –, amelyekből statisztikai alapon a szabályok és kivételek "generálhatók": a nagy elemkészleten működő eljárásokat lehet szabályként megfogalmazni, míg a kisebb halmazokon működők lesznek a kivételek. Ezzel a típusú leíró nyelvtannal kapcsolatos elképzeléseinket ismertettük a 2003-as MSZNY konferencián [4].

Jelen előadásunkban adatbázisunk egy újabb felhasználhatóságát mutatjuk be: egy "dinamikus korpusz" kidolgozását. Azért dinamikus, mert nem a *létező* (valaha létezett) alakokat tartalmazza, hanem az azokból visszaszármaztatott

elemeket és szabályokat, így az adott nyelvállapot *lehetséges* szavai, kifejezései vagy mondatai bármikor generálhatók – a kompetenciánkat modellálja tehát.

Mire lehet alkalmas egy ilyen korpusz? Többek között kutatások támogatására, ahogy a "hagyományos" korpuszok is [11]. Amiben az általunk dinamikusnak nevezett korpusz többet nyújt, az az, hogy tulajdonságra is kereshetünk benne. Kérhetjük például a programot arra, hogy generáljon főnévi igenevet tartalmazó mondatokat, vagy olyan szavakat, amelyekben több fonológiai váltakozás is van Így egyszerűen lehet majd egy-egy elmélethez példákat, illetve ellenpéldákat találni, vagy akár statisztikákat készíteni, hogy egy bizonyos tulajdonság (jegy) mennyire gyakori a nyelv lexikai egységeinek körében.

Egy konkrét számítógépes nyelvészeti programmal is együttműködünk: a szintén GASG-nyelvtanra épülő GeLexi-projekt [3] számára biztosítunk dinamikusan bővíthető lexikont. Ennek a nyelvtan formalizálhatóságát bizonyító szoftvernek az adatbázisa jelenleg a Prológ programnyelven íródott elemző-kód része, ezáltal benne bármiféle változtatást végezni nagyon nehézkes. Ezért szükséges, hogy egy korszerűbb adatbázist építve legyen biztosítható az említett implementációhoz a megfelelő lexikon. Az általunk kínált adatbázis a lexikai egységeket az összes tulajdonságuk-kal együtt tartalmazza, így egy egyszerű unifikációs eljárással (amely az egyetlen művelet a GASG-ben) az elemek szavakká, illetve mondatokká tudnak épülni. A program jelenleg is működik, de a lexikon elkészülte után sokkal nagyobb elemszámú korpuszon lesz képes ellenőrizni a szavak jólformáltságát (helyesírást), a mondatok grammatikalitását, és tud majd szemantikai reprezentációt társítani a beírt mondatokhoz.

Az épülő adatbázist teljes elkészültéig oktatási célokra kívánjuk hasznosítani. Két nagyobb alkalmazási területet különíthetünk el.

Az egyik a köz- és felsőoktatásban tanuló (magyar anyanyelvű) diákok nyelvtantanulásának segítése. Köztudott, hogy napjainkban a magyar nyelvtan tanítása igencsak elavultnak tekinthető, továbbá hiányzik belőle minden játékosság, problémaorientáltság, semmi nem motiválja a gyerekeket arra, hogy meg akarják ismerni anyanyelvük szabályszerűségeit. Úgy gondoljuk, egy olyan programmal, amivel gyakoroltatni lehet az egyes szabályokat, megvizsgálni működésüket azáltal, hogy "ki-be kapcsolgatjuk" őket, érdekesen lehetne megtanítani a diákoknak, hogyan működik a magyar nyelv. A program természetesen nem (csupán) a szabályok megtanítására és szemléltetésére lenne alkalmas, hanem a nyelvi tudatosság fejlesztésére is, ami a felsőoktatásban tanuló, később magyartanárként elhelyezkedő fiatalok számára lehet különösen fontos.

A másik fontos alkalmazási terület pedig a magyar nyelv tanítása idegen anyanyelvűeknek (hungarológia). Számukra talán még fontosabb, hogy lássák, hogyan, milyen szabályok szerint működik a magyar nyelv, illetve mely szavak vagy kifejezések viselkednek másképp. Emellett természetesen használhatják a programot arra is, amire a magyar anyanyelvűeknek nincs szükségük: megnézhetik, hogy az általuk "összerakott" szóalak vagy mondat helyes-e, és ha nem helyes, azt is láthatják, melyik szabályt nem alkalmazták, vagy használták rosszul. Ha pedig idegen nyelvű lexikai egységek is szerepelni fognak az adatbázisban, használhatják azt (n-nyelvű) szótárként. Ebben az esetben természetesen nem

csupán magyartanítás, hanem bármilyen más nyelv tanításának támogatására is alkalmas lesz a lexikonunk.

2 A megvalósítás módszerei

Célkitűzéseink megvalósításához alapfeltétel a rugalmasan bővíthető lexikon és szabályrendszer. Fontos, hogy a rendszerünk lehetőséget nyújtson a felhasználónak arra, hogy ne csak a lexikonban tárolt lexémákat tudja bővíteni, hanem ezekhez egy felhasználóbarát felületen szabályokat is tudjon rögzíteni, ami jelentheti akár teljesen új szabályok definiálását is. Továbbá lehetőséget akarunk adni a felhasználónak arra, hogy a szabályokat tesztelhesse: a különböző szabályok egymástól független ki-bekapcsolása alapján ellenőrizhesse, hogy grammatikus szóalakok, illetve mondatok generálódnak-e vagy sem, továbbá, hogy a felvett új szabályok nem generálnak-e túl.

A kiindulópontnak választott modell (GASG) olyan totálisan lexikalista grammatika, amelyben a fonológiai, morfológiai, szintaktikai, szemantikai szabályok egy szinten ábrázolódnak és időben egyszerre lépnek működésbe. Ezt a modellt vettük alapul a tervezett dinamikus korpusz megvalósításánál, és a célkitűzéseinknek megfelelően módosítottuk. Megtartottuk a GASG-nek azt az elképzelését, hogy a lexikonbeli egységek alapvetően morfémák, illetve néhol allomorfok, amelyeket egy külön jeggyel kapcsolunk össze morfémákká. Továbbá megtartottuk a GASG modellből a szórendre vonatkozó szabályok megadásának módját: a LiLe-ben is inegelőzési relációk, illetve ezeknek a rangparaméterezései fejezik ki a szükséges szórendi viszonyokat a mondatokban. A harmadik közös pont a mondatok szemantikai struktúrájának felépülése, amit szintén a GASG-ben definiált módon kívánunk megvalósítani.

A mi újításunk a GASG-hez képest az, hogy nem csak a különböző jegyek (feature) értékei, hanem maga a jegy-struktúra és a jegyekre épülő szabályok is adatbázisbeli elemek, vagyis nem a program (az elemző) részét képezik, hanem az adatbázisét. Így tetszőlegesen bővíthető a szabályok száma, és ez nem befolyásolja a program működését. Ezért valósítható meg az is, hogy a létező szabályok futását korlátozzuk vagy engedélyezzük, akár a lexémák tetszőleges halmazán. Ez természetesen némileg másképp definiált jegyeket és szabályokat tett szükségessé a GASG-hez képest. Szem előtt tartottuk azt is, hogy az általunk létrehozott rendszer ne legyen nyelv-specifikus, vagyis tetszőleges nyelv szabályait (és ennek alapját képező jegy-struktúráját) kódolni lehessen a rendszerünkben. A legfontosabb különbség a GASG és a LiLe között az, hogy míg a GASG egy grammatikai modell (mint az LFG vagy HPSG), és így mind szemléleténél, mind felépítésénél fogva általánosításokat fogalmaz meg az univerzális grammatikával kapcsolatban is (pl. régens és vonzatok lehetséges egyeztetési módjai stb.), addig a LiLe nem tartalmaz az univerzális grammatikára nézve ilyen típusú megállapításokat, csupán arra törekszik, hogy minden elképzelhető kapcsolat rögzíthető és ellenőrizhető legyen.

A jegyek és szabályok alkalmazását egy példán keresztül mutatjuk be. Vizsgáljuk meg a *lovakat* szóalak elemzését a programban! (Elemzéseink kiindulópontja [6] [7] volt.)

Ez a szó az elemzőnk szerint három morfémát tartalmaz: a *ló* szótövet, a főnévi többes számot kifejező morfémát, valamint a tárgyesetet kifejező morfémát. A szóalak grammatikus, mert a morfémák szófaja megegyezik, a morfémák a helyes sorrendben fordulnak elő, a *ló*-nak a megfelelő allomorfja szerepel benne, megfelel a hangrendi illeszkedés törvényének és a nyitásra vonatkozó hangtani szabályoknak is. Ahhoz, hogy ezeket megállapíthassuk, tárolni kell a szabályokat és az egyes morfémák és allomorfok tulajdonságait. A tulajdonságokat jegyekben tároljuk. Az egyes jegyeket el is neveztük, ám ez csak a szabályok megnevezéséhez, és így a felhasználó informálásához szükséges. A következő leírásban az egyes jegyek megnevezését adjuk meg, de a programban valójában csak a jegy azonosítójára hivatkozunk, a jegy nyelvészeti tartalma (szófajok, fonológiai tulajdonságok stb.) a program működése szempontjából irreleváns. Ezt azért tartjuk fontosnak megjegyezni, mert az egyes jegyek elnevezésénél nem törekedtünk arra, hogy valamely konkrét grammatikai (a példánkban: fonológiai) modell terminológiájához illeszkedjünk. A terminológia végső kialakítása az adatbázisunkat alapul vevő tananyagok része lehet.

A *ló* tulajdonságai közé tartozik, hogy főnévi tő, ami *ló* és *lov* alakokban fordulhat elő; továbbá mély hangrendű. A lov allomorf (a megjelölt toldalékok esetében) kötőhangot kíván meg; ún. nyitótő, ami befolyásolja az őt követő kötőhangot; valamint ún. v-vel bővülő tő. A többes szám jele (*-k*) tulajdonságai közé tartozik, hogy névszói toldalék, ami közepes erősséggel akar a szótőhöz közel kerülni; kiváltja a v-vel bővülést; és a következő morfémára vonatkozóan ún. nyitótőként viselkedik. A lehetséges allomorfjai közül az *-ak* mély hangrendű és a nyitótövekhez társul. A tárgyeset ragjának a példánkban releváns tulajdonságai: névszói toldalék; kis erősséggel akar közel kerülni a tőhöz; az *-at* allomorfja pedig mély hangrendű és nyitó "tövekhez" (vagyis megelőző morfémákhoz) társul. A jegyek listájában megadjuk, hogy mely jegyek párosíthatóak, illetve azt is, hogy melyek zárják ki egymást, így egyfajta unifikációs módszerrel ellenőrizhető, hogy a szóalakban szereplő morfok jegyei illeszthetőek-e. Természetesen nem minden helyzet ilyen egyértelmű, ekkor szintén adatbázis-rekordként tárolt, bonyolultabb szabályok lépnek működésbe az egyszerű jegy-unifikáció helyett.

Az ismertetett lexikon- és jegy-struktúrát SQL-adatbázisban tároljuk, és az elemzéskor építünk az SQL-szerver gyors keresést biztosító működésére. A jelenlegi programverzió objektumorientált nyelven (Delphi) íródott, mivel ezzel tudjuk legkönnyebben biztosítani a felhasználóbarát felületet, de a későbbiekben terveink között szerepel a webes megvalósítás, aminek alapjául az szolgál, hogy az adataink akár online kinyerhetőek xml-formátumban.

Az adatbázis feltöltésének két fázisa van. Jelenleg, az első fázisban a feltöltést kézi módszerrel végezzük, és eközben bővítjük a jegy-struktúrát és a szabályrendszert. A feltöltés a magyar nyelv fonológiájának – morfológiájának oktatása közben történik, egyetemi hallgatók bevonásával. A második fázisban (egy éven belül) áttérünk az automatizált feltöltésre: programunk jelenleg is alkalmas szövegek szavakra bontására és az ismert szavak elemzésére. Kidolgoztunk egy eljárást az ismeretlen szótövű alakok morfémáinak jóslására és a rendelkezésre álló adatok alapján a jegyekkel való ellátására is. Természetesen ezeket a géppel létreho-

zott adatokat ellenőrizni kell, erre a legalkalmasabb a belőlük generált szóalakok és mondatok helyességének ellenőrzése: ekkor ugyanis az ellenőrzésbe nem csak nyelvészek vonhatóak be, hanem laikus magyar anyanyelvű felhasználók is.

3 Amiben újat nyújtunk

Számos számítógépes nyelvészeti projekt dolgozik azon, hogy valamilyen alkalmazást, szótárt fejlesszen, elméleti (kutatási, pl. korpuszok, szótárak) vagy gyakorlati (pl. helyesírás-ellenőrzés, fordítás) célokra. Ezekhez a projektekhez képest mi a célkitűzéseinkben, a felhasznált elméleti keretben, az alkalmazott technológiában és a működésben tudunk különböző szempontokból újat nyújtani.

A LiLe célja nemcsak a kutatások támogatása, illetve nem elsődleges célunk helyesírás-ellenőrző vagy fordító-programok kifejlesztése, ahogy a legtöbb számítógépes nyelvészeti projekt esetében, hanem az egyik speciális célkitűzésünk az eredmények taneszközként való felhasználása a nyelvoktatásban. A nem anyanyelvi beszélők hiányzó kompetenciája pótolható, kisegíthető azáltal, hogy egyes morfémasorok helyes vagy helytelen voltát el tudja dönteni programunk. Ennek alapja pedig nem karakteregyezés (tárolt vagy generálással előállított elemekkel), hanem azok a szabályok, melyek az egyes elemeken működnek. Ebből kifolyólag a döntés alapjául szolgáló elveket is meg tudja nevezni, amely hasznos segítség lehet a nyelvoktatásban, illetve a nyelvi tudatosság fejlesztésében. Mivel szabályainkat – az elemekkel együtt – adatbázisban tároljuk, egymástól függetlenül kijelölhető az elemzendő elemek halmaza és az azokon működő szabályok köre is, amely az egyes nyelvi jelenségekkel kapcsolatos gyakorlásra, szemléltetésre ad módot.

A felhasznált elméleti keret kiválasztásánál sem az általánosan elterjedt frázisstruktúra-nyelvtan valamely modellje mellett döntöttünk. Mivel nemcsak a tárolt elemeket, hanem az azokon működő szabályokat is ugyanazon eszközrendszerrel kezeljük, legyen szó a nyelv bármely "szintjéről", egy totálisan lexikalista modellt választottunk elméleti háttérül, amely ezt biztosítani tudja. Az, hogy a korpuszunkban morfémákat tárolunk kész szavak helyett, nem példa nélküli (noha nem is általános) a számítógépes nyelvészeti alkalmazások körében. Továbbá az sem, hogy a nyelvtant nem szabályhalmazként képzeljük el, ebben ugyanis követjük az unifikációs modelleket, amelyekben a "szabályok" nem igazi szabályok önmagukban, hanem az egyes elemek tárolt tulajdonságainak (jegyhalmazainak) egyeztetésén alapulnak. A mi újításunk a megszokott unifikációs modelleknél (pl. HPSG) is szigorúbban lexikalista GASG kiválasztása, és annak a még erőteljesebb "lexikalizálása" az elemző-kód megszüntetésével, és minden szabálynak az adatbázisba való beépítésével. A morfémák tárolása a kész szavak helyett lehetőséget ad a speciális célunk (oktatás) támogatására, valamint a fő célkitűzésünk (dinamikus korpusz: összes lehetséges alak generálása) megvalósítására. A minél teljesebb lexikalizmus is ezt a célt szolgálja: az adatbázisunk szabad bővítésével vagy változtatásával a módosítások a "nyelvtan" működését is meghatározzák, így a felhasználó (tanuló vagy kutató) gyorsan és hatékonyan tudja tesztelni a nyelvi kompetenciáját vagy a nyelvészeti modelljének működését.

Az alkalmazott technológia megválasztásánál a fő szempontunk az volt, hogy olyan technológiát válasszunk, melyben a sok különböző elem és összefüggéseik is egyféleképpen tárolhatóak, különbségeiket mégis megtartva. A relációs adatbázis ad erre lehetőséget. Továbbá az általunk használt MS-SQL-be beépített eljárások arra is módot adnak, hogy az általában tárolási célra használatos xml-t előállítsuk. Erre elsősorban a webes megjelenítésnél lesz szükség.

Mind a választott elméleti keret, mind az alkalmazott technológia lehetőséget ad a működésbeli újításra. A LiLe adatbázisában, azon túl, hogy az egyes morfémák külön-külön rekordként szerepelnek, ugyanilyen formában tároljuk az egyes, jegyekkel definiált tulajdonságokat is, amelyek vagy unifikációs eljárások vagy "szabályok" bemenetéül szolgálnak. A morfémák tárolásánál különböző változókat használunk az allomorfok elkülönítésére, kiszámítására. Ezek a változók is külön rekordok, melyek vagy az unifikációt szolgálják ki, vagy – szintén adatbázisban tárolt – eljárásokat hívnak meg az unifikációs módszerrel nem elemezhető jelenségek esetében. Ezek az eljárások műveleteket végeznek a tárolt elemeken. Mivel a változók a morfémák részei, azt a kört is pontosan definiáljuk, hogy hol jelennek meg a nyelvben az adott jelenségek. Ilyen értelemben a nyelvekben gyakran jelentkező "kivételek" kezelése sem jelent problémát, hiszen nem kell kivételeket tárolnunk – egy adott jelenség ilyen esetben egy kis elemszámú halmazon működik.

A felhasznált elméleti keret és annak általunk végrehajtott módosításai, valamint az alkalmazott technológia mind támogatják azt a már megfogalmazott célkitűzésünket, hogy egy olyan általános keretet tudjunk biztosítani, amely lehetőséget nyújt a grammatikai szabályok akár nyelvenként eltérő megfogalmazására. Vagyis nem célkitűzésünk egy univerzális grammatika létrehozása (szemben például a GASG törekvéseivel). A LiLe-ben az egyes nyelvek fonológiai/morfológiai/szintaktikai leírásában szereplő szabály-struktúrát az adott nyelv leírói határozzák meg, és az egyes nyelvtanok a közös struktúrájú szemantikai reprezentáción keresztül kapcsolódhatnak egymáshoz. Vagyis a lexikai egységek jellemzésében csak a szemantikai jegyek közösek (univerzálisak), a többi jegy szabadon alakítható. Így kívánunk bármely grammatikai modell számára hasznosítható lexikont biztosítani.

4 Eddigi eredmények

Kidolgoztuk azt a relációs adatbázis-struktúrát, amely minden további már megvalósult és jövőben megvalósuló elképzelésünknek az alapja.

A felépített struktúrának köszönhetően megvalósíthatóvá vált a totális lexikalizmus elve. Minden egyes lexikai elemet külön tételként, rekordként szerepeltetünk az adatbázisban, nem az összes előfordulási alakjában, hanem a tőmorfémákat és a toldalékokat külön, az összes releváns tulajdonságukkal együtt. A lexikai egységek mellett az adatbázisban kaptak helyet a lexikai egységek viselkedését befolyásoló szabályok is, tehát szabályaink is a lexikonunk részét képezik, ahogy ezt a korábbiakban már részletesen ismertettük.

A kidolgozott struktúra és a technológia lehetőségei együtt szinte korlátlan bővíthetőséget biztosítanak. Bővülhet nem csak a hagyományos értelemben vett lexikon, hanem a nyelvtan maga is: gond nélkül felvehetünk újabb értékeket, tulajdonságokat mind a lexikai egységekhez, mind a tulajdonsághalmazunkhoz, bármilyen nyelven. A munka jelenlegi szakaszában, a korábban definiált első fázisú adatfeltöltési módszerünkkel egyidőben változott és változik az adatbázis, méghozzá az alapötlet megsértése vagy változtatása nélkül.

Már a munka kezdetekor célul tűztük ki magunk elé, hogy az adatbázist úgy tervezzük meg és hozzuk létre, hogy az ne csak a saját fejlesztésünk alapja lehessen, hanem más nyelvészeti elmélethez és gyakorlati megvalósuláshoz is hasznos segítséget nyújthasson, vagy azoknak is alapjául szolgálhasson. Az építkezés során mindig a szemünk előtt lebegett egy olyan elméletektől független, vagy legalábbis kevéssé befolyásolt lexikon képe, amely többféle próbálkozáshoz, elképzeléshez tud kapcsolódni. Ennek egyik fontos technológiai alapja lehet a rendelkezésünkre álló XML-adatszolgáltatás.

Az adatbázis mellett a projektünk eddigi eredményei közé tartozik egy szoftver is, amely mostanra két egymástól élesen elkülönülő funkcióval rendelkezik.

Az egyik funkció az adatbázis feltöltése. Az adatrögzítést azzal segíti a szoftverünk, hogy a táblastruktúra részletes ismerete nélkül is bővíthető az adathalmaz. Ezt a feltöltő programot a jövőben weben elérhető formában, autentikációval kibővítve fogjuk a felhasználók rendelkezésére bocsátani. Az autentikáció előnye, hogy a nyelvtani rendszeren végrehajtott változtatások vagy a lexikon bővítései felhasználókhoz köthetőek.

A szoftver másik része egy szóelemző program, amely jelen állapotában a magyar főnévi és igei inflexiós morfológia és az írásban jelölt fonológiai szabályok alapján működik. Figyelembe veszi a szófaji egyezést, a helyes morfémasorrendet és a fonológiai szabályokat. Az egyedisége abban rejlik, hogy nemcsak azt tudja megállapítani, hogy a beírt szóalak helyes-e, hanem azt is megmondja a felhasználónak, hogy mely nyelvtani szabály helytelen működése vagy figyelmen kívül hagyása eredményezte a nem megfelelő szóalakot. Úgy gondoljuk, hogy ennek köszönhetően a programunk továbbfejlesztett változata igen hathatós segítséget nyújthat a külföldiek magyartanításában és a magyar anyanyelvűek nyelvi tudatosságának fejlesztésében.

A projekt már jelenlegi fázisában is az oktatás részét képezi, hiszen az elmúlt félévben a PTE BTK Nyelvtudományi Tanszékén egy kutatószeminárium keretében a hallgatóink megismerhették az adatbázis felépítését és működésének elveit. Miközben az adatbázis releváns adatokkal való feltöltésével a magyar morfofonológia tanulásához gyűjthettek gyakorlati tapasztalatokat, a mi munkánkat is segítették egy-egy remek ötlet felvetésével.

5 Jövőkép

Az adatbázis kidolgozásán túl az elsődleges célunk a lexikon feltöltése magyar nyelvi adatokkal, és ezáltal egy leíró magyar nyelvtan definiálása.

Legközelebbi céljaink közül az első nyelvészeti indíttatású. Az adatbázis-struktúra morfológiai bővítésén dolgozunk, a morfológiai szabályrendszert kívánjuk kiterjeszteni a képzők körére is. Számítógépes nyelvészeti feladat az adatbázis automatizált feltöltésére való áttérés a már kidolgozott eljárási elveink alapján. Tisztán informatikai feladat az elkészült program webes környezetbe való átültetése, amely szélesebb körben teszi elérhetővé a megvalósuló eredményeket.

További céljaink között kívánjuk megvalósítani a nyelvészeti feladatok terén a magyar szintaxis kidolgozását a modellünkön belül. A számítógépes nyelvészeti munka területén annak a már bemutatott dinamikus korpusznak a megvalósítását tervezzük, amelyhez minél több elméleti és gyakorlati kutatás kapcsolódhat. Végül pedig ugyancsak egy-két éven belül tervezzük létrehozni azt az oktatástámogató programcsomagot, amely a hungarológiaoktatás és a közoktatás segítségére lehet.

A távlati célok közül az egyik legfontosabb az n-nyelvűség kérdésének gyakorlati megvalósítása, a másik pedig egy olyan lexikalista szemantika felépítése, amelyet adaptálni tudunk a már meglévő adatbázis-struktúránk keretei közé.

References

1. Alberti Gábor (1998): *GASG: Minimal Syntax, Maximal Lexicon and PROLOG*, In: Hunyadi László szerk.: ALLC/ACH '98. KLTE, Debrecen; 81-83.
2. Alberti Gábor (1999): *GASG: The Grammar of Total Lexicalism*; In: Working Papers in the Theory of Grammar 6/1, Elméleti Nyelvészet Program, ELTE és MTA Nyelvtudományi Intézet.
3. Alberti Gábor – Balogh Kata – Kleiber Judit – Viszket Anita (2002): *A totális lexikalizmus elve és a GASG nyelvtan-modell*; In: Maleczki Márta szerk.: A mai magyar nyelv leírásának újabb módszerei V. Szegedi Tudományegyetem; 193-218.
4. Bódis Zoltán – Kleiber Judit – Szilágyi Éva – Viszket Anita (2003): *Leíró nyelvtan – adatbázisból*; In: Magyar Számítógépes Nyelvészeti Konferencia MSZNY2003 Szeged, 2003 december 10-11. Konferenciakötet, SZTE; 300-302.
5. Bódis Zoltán – Kleiber Judit – Szilágyi Éva – Viszket Anita (2003): *Nyelvészeti lexikon – oktatási és kutatási adatbázis fejlesztése*; Konferencia-előadás: Multimédia az oktatásban, Pécs, 2003. október 10.
6. Kiefer Ferenc szerk. (1994): *Strukturális magyar nyelvtan 2. Fonológia*; Bp. Akadémiai Kiadó.
7. Kiefer Ferenc szerk. (2000): *Strukturális magyar nyelvtan 3. Morfológia*; Bp. Akadémiai Kiadó.
8. Mitkov, Ruslan szerk. (2003): *The Oxford Handbook of Computational Linguistics*; Oxford University Press.
9. Prószéky Gábor – Kis Balázs (1999): *Számítógéppel – emberi nyelven. Intelligens szövegkezelés számítógéppel*; Szak Kiadó, Bicske.
10. Prószéky Gábor – Olaszy Gábor – Váradi Tamás (2003): *Nyelvtechnológia* In: Kiefer Ferenc szerk.: *A magyar nyelv kézikönyve*, Akadémiai Kiadó; 567-589.
11. Svartvik, Jan szerk. (1992): *Directions in Corpus Linguistics*, Proceedings of Nobel Symposium 82 Stockholm, 4-8 August 1991; Mouton de Gruyter, Berlin – New York.

Word Order and Discontinuities in a Dependency Grammar for Hungarian

Csongor Barta[1], Ricarda Dormeyer[2], and Ingrid Fischer[2]

[1] Computer Science Department, University of Szeged, 6000, Kecskemét, Katona Zsigmond u.22., Hungary
barta.csongor@stud.u-szeged.hu
[2] Lehrstuhl für Informatik 2, Universität Erlangen-Nürnberg, Martensstr. 3, 91058 Erlangen, Germany
{Ricarda.Dormeyer, Ingrid.Fischer}@informatik.uni-erlangen.de

Abstract. Natural languages are always difficult to parse. Two phenomena that constantly pose problems for different formalisms are word order—what part of a sentence has to be placed where—and discontinuities—words that belong together but are not placed into the same phrase. Dependency grammar, a linguistic formalism based on binary relations between words, is very adequate for handling both problems. A parser for dependency grammar together with its grammar writing formalism is described in this paper. Word order and discontinuities in Hungarian are handled based on this formalism.

1 Introduction

At the Computer Science Department of the Friedrich-Alexander University Erlangen-Nuremberg a lot of different parsers have been developed over the years. One of the newest developments, a parser for dependency grammar [4], is currently tested for several different languages. Grammar fragments for English, German and Latin have been written [4]. These languages differ in their word order. English has a fixed word order, e.g. the subject has to come first in a declarative sentence. In German, the word order is semi-free. For noun phrases, it is fixed; on the sentence level, the verb has to be in the second position in a declarative sentence. Other elements can take the other positions, no restrictions are given here. In Latin there are even less word order restrictions. Words can be placed nearly everywhere.

But not only word order is of special interest. Another problem are words that belong together but are not placed next to each other in the sentence. This phenomenon can be found in all languages analyzed. An English example is fronting as in *Ann John told me he had seen.* In German and Hungarian verb prefixes can be separated from the verb and move to another position.

Now non-indo-European languages are researched. Currently grammars for Japanese [17] and Hungarian are developed. For Japanese, a lot of different

Fig. 1. A dependency tree for *János keresi Marit*

dependency grammar implementations exist. This is not the case for Hungarian. Only one international publication could be found containing a dependency grammar for Hungarian [18]. The grammar described in [18] differs a lot from our approach. In [18] first all prefixes and suffixes are separated from word stems. The resulting string is the input for the dependency parser. In our approach this separation does not take place, a sentence is analyzed in its original writing.

In the sequel, our dependency parser and the Hungarian grammar developed up to now are described. In section 2 the basics of dependency grammar are introduced. After this linguistic introduction, our dependency parser is specified in section 3. Special features of our Hungarian grammar are given in section 4. We end with a conclusion.

2 A Short Overview on Dependency Grammar

When taking a look in the standard literature [9] on computational linguistics, long introductions in phrase structure grammars invented by Chomsky can be found. They have been in the focus for nearly forty years now. Phrase structure grammars turned out to be a helpful method when modeling English; quite a lot of parsers can be found together with extensive grammars. But it also turned out that they are not useful when it comes to languages with free or semi-free word order. Discontinuous constituents and long distance dependencies pose difficulties, too. Several work arounds and extensions have been invented to overcome these problems. Some of these extensions and new developments, e.g. *Head-Driven Phrase Structure Grammar* [13], are similar to dependency grammar. Dependency grammar, invented by Lucien Tesnière [14] [15], is popular in Europe and Japan. In this theory binary relations between the words of a sentence are used as the basic construct. The most important part of a sentence is the verb, it opens several slots for other parts of the sentence. Taking the verb *keresi*, it opens two slots, one for a noun in the nominative case, which is the subject, and one for a noun in the accusative case, the object. In the sentence *János keresi Marit* these slots are occupied by *János*, the subject, and *Marit*, the object. Normally the relations are visualized with the help of trees. A tree for the running example is given in Figure 1. Please note, that every combination of the three words results in the same dependency tree: *János Marit keresi*, *Marit keresi János*, ...

The subject and object can also open new slots. A simple noun opens a slot for e.g. one determiner and any number including zero of adjectives. This means that several different kinds of slots are necessary. First slots are used that must

Fig. 2. A dependency tree for *Meg kell, hogy beszéljük a problémákat*

be filled, with one element. If the element is missing, the corresponding sentence is grammatically not correct. In Figure 1 exactly one object is needed. In English each verb needs exactly one subject. Then there are slots that are optional, they can be filled but do not have to be filled. Time and place are optional for most verbs. They can be added, but they can also be left out. Another example for this is the subject in Hungarian. Finally there are slots that can be filled several times, e.g. a noun can take several adjectives. A word opening a slot is also called the head, the word filling the slot will be called the dependent in the rest of the paper.

A more complicated sentence as *Meg kell, hogy beszéljük a problémákat* leads to the tree in Figure 2. This sentence contains a discontinuous constituent, another popular problem when analyzing natural languages. The word *meg* is followed by a word of the main clause, but syntactically it belongs to the verb of the subordinate clause *beszéljük*. In the tree in Figure 2 *meg* is drawn as a dependent of *beszéljük*. The linear structure of the words in this sentence cannot be reconstructed from Figure 2. In the tree only the syntactic structure is given. In phrase structure grammar linear and syntactic structure are combined in one tree leading to crossing edges in the phrase structure tree for this sentence. Trees with crossing edges cannot be constructed with context-free grammars.

3 A Parser for Dependency Grammar

Our parser [16] is based on three concepts. The parsing algorithm itself is similar to the well-known Cocke-Kasami-Younger algorithm for context-free phrase structure grammars [9]. The first dependency parser based on this idea was presented in [12]. Words are described by feature structures [9] enriched by a few symbols necessary for dependency grammars. Feature structures are combined with the help of graph unification. Our handling of discontinuous constituents and word order restrictions differs from [12].

3.1 Word Order

In Tesnière's original approach, word order was unimportant for syntactic description. Any order was allowed. This is not useful for parsers, too many wrong

sentences would be accepted. The order between head and dependent must be considered as well as the order between the different dependents of one head. Also the number of elements following or preceding a word can be important. E.g. in German the verb in a declarative sentence must fill the second position. In our approach each word has a position list where positions of the word itself and other words are described. This includes as a minimum a position for the word itself. Then each dependent of a word can have a fixed position in this position list. Free positions within the position list are also possible, a free position can take every dependent that is not marked as fixed. The position list makes parsing easier: When a fixed position is following according to this list, the parser has to check for just one element. If the next element is free, only free elements have to be checked.

3.2 Discontinuous Constituents

Linguistically, a discontinuous dependency can be regarded as a ternary relation, i.e. a relation between a dependent, its *syntactic head* and its *positional head* or linear head [1]. The syntactic head is the word containing a slot for the discontinuous dependent. constituent. But because the syntactic head's constituent is discontinuous, the dependent is positioned in the position list of the linear head. In the example sentence, the syntactic head for *meg* is *beszéljük* and its linear head is *kell*. The dependent fills a slot of its syntactic head, but occupies a position of the linear head's position list. Because of this, the processing of discontinuous dependencies starts with the linear head. When the parser finds the syntactic head, it attaches its unfilled slots to the linear head; when it finds the dependent, it allows the dependent to occupy a position in the linear head's position list while not yet filling a slot. At the end of the parsing of the linear head's position list, the open slots are filled by these dependents. A similar approach to parsing dependency structures can be found in [19].

3.3 Other Approaches

Over the years several other parsers for dependency grammar have been proposed. In [1] linear order and syntactic order are strictly separated, something we tried to avoid in our approach, because it makes grammar writing less intuitive and parsing less efficient. Fraser's parser [5] is based on backtracking and uses a parsing stack. Covington's approach [2] is cited very often. He invented a simple backtracking algorithm for free word order. But his approach is not well suited for semi-free word order phenomena. Finally there are several dependency parsers based on constraint resolution, a completely different approach [3].

4 Parsing Hungarian

In this section two Hungarian sentences are analyzed. With these examples our grammar description language is described.

```
Word "Angéla" <"Name"> [
    lexeme: Angéla;
    gender: fem;
    case: nom;]

Word "olvas"   <"VerbPres"> [
    lexeme: olvas;
    mood: declarative;
    number: sing;
    person: 3;]

Template "Name" [
    category: noun;
    special: propername;
    number: sing;
    person: 3;]
```

```
Template "VerbPres" [
    category: verb;
    form: finite;
    tense: present;
    sentence: declarative;
    subj:  oslot [
        category: noun;
        cont: +;
        case: nom;
    ];
    order: (%1 %2 i %3);
]
%1 = slot  [cont: +;];
%2 = mslot [cont: +;];
%3 = mslot [cont: +;];
```

Fig. 3. Simple grammar with templates for *Angéla olvas.*

4.1 Grammar Description Language and Free Word Order

The grammar description language is important, as it must be easy to learn and to handle for the linguist writing grammars for the parser. We will introduce it with the help of the easy example *Angéla olvas.* In Figure 3, the corresponding grammar is given. Please note that due to space our example grammars are not complete.

To shorten the grammar, templates as introduced by [6] are possible. Templates encode parts of the feature structures that are used very often. Within the entries for words template names are used instead of complete feature structures. As a first step, the lexicon is transformed before parsing. Template names are removed, the feature structure parts these names described are unified with the rest of the feature structure.

Feature structures are started by "[" and ended with "]", features and values are separated by ":" and feature-value-pairs are separated by ";". In Figure 3 two word entries and two templates are given. The lexical entry for *Angéla* contains a template named `Name`, which is also given. Feature structures for `Name` and *Angéla* are unified leading to an entry where agreement features as `number`, `gender`, `case` (not all possible cases are given) and `person` are described. Also the `lexeme` and `category` are shown. The verb *olvas* is also composed with the template for verbs in present tense. This template is more interesting. It contains an optional slot for a subject indicating that in Hungarian, the subject can be left out. At the end the special feature `order` indicates possible positions. As this position list is used for every word in present tense, it is more complicated than necessary for our small example. The symbol `i` stands for the position of the word itself, in this case the current verb. Before this verb at least one element must be placed. %1 must be a slot, this slot must be filled. %2 is marked `mslot` (multiple

```
Word "kell" [
    category: verb;
    lexeme: kell;
    subj: slot  [
        category: subjunction;
        lexeme: hogy;
        cont: +;];
    order: (%1 i %2);]
%1 = mslot;
%2 = mslot [cont: +;];

Word "beszéljük" [
    category: verb;
    lexeme: megbeszélni;
    subj: oslot [category: noun;];
    obj:  slot  [category: noun;];
    verbpart: slot [
        category: verbpart;
        lexeme: meg;];
    order: (%1 i %2);
]
%1 = mslot [cont: +;];
%2 = mslot [cont: +;];

Word "meg" [
    category: verbpart;
    lexeme: meg;]

Word "problémákat" [
    category: noun;
    lexeme: probléma;
    spec: %1 oslot [
        category: determiner;
        cont: +;];
    order: (%1 i);]

Word "a" [
    category: determiner;
    lexeme: a;]

Word "hogy" [
    category: subjunction;
    lexeme: hogy;
    prop: %1 slot [
        category: verb;
        cont: +;]
    order: (i %1);]
```

Fig. 4. Simple grammar for *Meg kell, hogy beszéljük a problémákat.*

slot). A multiple slot can be filled with an arbitrary number of elements but can also be empty. An arbitrary number of elements can also follow after the verb. It can also be described that the subject must go in the first position; in this case %1 has to be added to the subject slot. Please note that this is a lexicalised grammar, i.e. all information is stored in the lexicon, no extra grammar rules are needed.

4.2 An Example With a Discontinuity

Our treatment of free word order has already been introduced in the previous section. Now a more complicated example taken from [11] is used to show how discontinuities are handled. The dependency tree has already been shown in Figure 2. Discontinuity is quite common in Hungarian especially in the tenses and the verbal prefixes, e.g. *Szilvia el akar menni, Angéla akar elmenni.* In Figure 4 a short and not complete grammar for the sentence *Meg kell, hogy beszéljük a problémákat* is given. Templates are left out for simplicity. In the grammar, continuity and discontinuity are marked using special features. A feature `cont` is used to specify whether a dependency may be realized only continuously ("+"), only discontinuously ("-"), or both (not specified); a second feature `cont-const` is used to specify whether dependents may be extracted from the constituent

headed by a certain word[3]. Both features can be applied to lexical entries of words, to slots, or to positions in a position list. Specification of dependents, slots or positions as continuous will stop this process from taking place.

To parse the sentence for example 4, the parser first encounters the words *meg* and *kell*. *Meg* contains neither slots nor a position list and therefore cannot act as a head. The verb *kell*, on the other hand, has no slot for a `verbpart` dependent, but it has an `mslot` position to its left which may also be filled by a discontinuous dependent of one of its dependents. Therefore, a new analysis is generated where *meg* fills this position on the assumption that its syntactic head will turn out to be a word depending directly or indirectly on its positional head *kell*. The third word *hogy* is read but not yet linked to its head *kell*, since it has itself an obligatory continuous slot that must be filled first. The same applies to the next word *beszéljük*. Only after both *a* and *problémákat* have been read is the first slot filled: the specifier (`spec`) slot in *problémákat* can be filled by *a*. After that *problémákat* has no open slot or position left and can itself fill the object (`obj`) slot and part of the %2 position of the subordinate clause's verb. Now the verb has two slots left. The subject slot is optional, and the slot for the verbal prefix may be discontinuous. Therefore, the verb can be bound to its head *hogy*. After that *hogy* can be linked to its head. The `verbpart` slot is passed first to *hogy* and then to *kell*. Now, finally, the word *meg* that has been bound only positionally is assigned to its slot, namely to the discontinuous `verbpart` slot.

5 Conclusion and Future Work

It is most important to add a Hungarian morphology to the parser. Up to now, the parser works with a full form lexicon for Hungarian. This might be a solution for other languages, but it does definitely not work for Hungarian due to the high number of possible word endings.

Word order phenomena and discontinuity in Hungarian were modeled for a dependency parser. It turned out that, as for the other languages tested, it was possible and easy to write down. Nevertheless there is still a lot of work to do. Up to now only special problems of Hungarian have been modeled as a proof of concept for the parser. Next a full-flexed Hungarian grammar should be developed.

References

1. Norbert Bröker, *Separating surface order and syntactic relations in a dependency grammar*, in Proceedings of the 36th Annual Meeting of the ACL and 17th International Conference on Computational Linguistics, Montreal, 1998.
2. Michael A. Covington, *Parsing discontinuous constituents in dependency grammar*, Computational Linguistics, 16(4):234-236, 1990.

[3] No example for `cont-const` is given.

3. Ralph Debusmann, Denys Duchier and Geert-Jan Kruijff *Extensible Dependency Grammar: A New Methodology*, Recent Advances in Dependency Grammar, COLING 2004.
4. Ricarda Dormeyer, *Syntaxanalyse auf der Basis der Dependenzgrammatik*, PhD Thesis, Computer Science, Friedrich-Alexander University Erlangen-Nuremberg, 2004.
5. Norman M. Fraser, *Parsing and dependency grammar*, UCL Working Papers in Linguistics, 1:296-319, 1989.
6. Peter Hellwig, *Chart parsing according to the slot and filler principle*, in Proceedings of the 12th International Conference on Computational Linguistics, pages 242-244, Budapest, 1988.
7. Richard Hudson, *Word Grammar*, Blackwell, Oxford, 1984.
8. Richard Hudson, *Towards a computer-testable word grammar of English*, UCL Working Papers in Linguistics, 1:321-338, 1989.
9. Daniel Jurafsky and James H. Martin, *Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Prentice Hall, New Jersey, 2000.
10. László Keresztes, *Hungaro Lingua: Praktische ungarische Grammatik*, Debreceni Nyári Egyetem, 1999.
11. Katalin Kiss, *The Syntax of Hungarian*, Cambridge Syntax Guides, Cambridge University Press, 2002.
12. Michael C. McCord, *Slot grammar. A system for simpler construction of practical natural language grammars*, in Rudi Studer, editor, Natural Language and Logic, pages 118-145. Springer, Berlin, Heidelberg, 1990.
13. Ivan Sag, Thomas Wasow and Emily Bender: *Syntactic Theory. A Formal Introduction*, Second Edition, Stanford: Univ. of Chicago Press, 2000.
14. Lucien Tesnière, Esquisse d'une syntaxe structurale, Klincksieck, Paris, 1953.
15. Lucien Tesnière, Eléments de syntaxe structurale, Klincksieck, Paris, 1959.
16. Thomas Tröger, *Ein Chartparser für natürliche Sprache auf der Grundlage der Dependenzgrammatik*, Master Thesis, Computer Science, Friedrich-Alexander University Erlangen-Nuremberg, 2003.
17. Alexandra Pröll, *Eine Dependenzgrammatik für das Japanische*, Bachelor Thesis, Computer Science, Friedrich-Alexander University Erlangen-Nuremberg, 2004.
18. Gábor Prószéky, Ilona Koutny and Balázs Wacha, *A dependency syntax of Hungarian*, in Dan Maxwell and Klaus Schubert, eds., Metataxis in Practice, pages 151-182. Foris Publications, Dordrecht, 1989.
19. Markus Schulze, *Ein sprachunabhängiger Ansatz zur Entwicklung deklarativer, robuster LA-Grammatiken*, PhD Thesis, Computer Science, Friedrich-Alexander University Erlangen-Nuremberg, 2004.

Kísérlet magyar szavak jelentéshasonlóságának meghatározására a *Magyar szókincstár* segítségével

Bárdosi Vilmos[1], Kiss Gábor[2], Kiss Márton[3], Rapcsák Tamás[4]

[1] ELTE BTK Francia Tanszék, tanszékvezető egyetemi tanár;
1088 Budapest, Múzeum krt. 4/c.;
e-mail: vbardosi@ludens.elte.hu
[2] MTA Nyelvtudományi Intézete, Korpusznyelvészeti Osztály; TINTA Könyvkiadó,
http://tintakiadó.hu, igazgató, főszerkesztő; 1117 Budapest, Kondorosi út 17.;
e-mail: kissgabo@tintakiado.hu
[3] SZTE Műszaki Informatika Szak, egyetemi hallgató; TINTA Könyvkiadó,
1117 Budapest, Kondorosi út 17.;
e-mail: Kiss.Marton.1@stud.u-szeged.hu
[4] MTA SZTAKI Operációkutatás és Döntési Rendszerek Laboratórium és Osztály,
tudományos osztályvezető; 1111 Budapest, Kende u 13–17.;
e-mail: rapcsak@oplab.sztaki.hu

Kivonat: A szavak jelentéshasonlóságának meghatározására irányuló kutatások és kísérletek a mintegy fél évszázados asszociációs pszicholingvisztikai kísérletek után az utóbbi évtizedben ugrásszerűen megnőttek. A növekedés okai a természetes nyelvek gépi feldolgozása technológiájának látványos fejlődése és a ma már széles körben elérhető elektronikus nagy nyelvi adatbázisok (egynyelvű szótárak, tezauruszok, korpuszok, WordNet) létrehozása. Előadásunkban bemutatjuk kísérletünket, melyben a *Magyar szókincstárat* [Kiss 1998], pontosabban az abban lévő 25787 címszó alatt található 42976 szinonimasort miként használtuk fel kiindulási nyelvi tudásbázisként szópárok (egyes aljelentések szerint megkülönböztetett) jelentéshasonlóságának meghatározására. Ismertetjük a szópárok jelentéshasonlósági mérőszámaiból felépített – szófajokra szétbontott – jelentéshasonlósági mátrixok létrehozásának menetét. Kísérletet végeztünk, hogy a jelentéshasonlósági mátrixokból kiindulva szinguláris érték dekompozíció (SVD) alkalmazásával miként lehet automatikusan fogalomköröket generálni.

1. A nyelvi intuíció és a szavak jelentésének hasonlósága

Közhelyszerű tény, hogy míg egyes szavak jelentése közel van egymáshoz, más szavaké távolinak tűnik. Az anyanyelvi beszélő a *ballag ~ sétál* szavak jelentését igen hasonlónak érzi egymáshoz, ugyanúgy mint a *fut ~ szalad* szópár tagjainak a jelentését is. Azonban a nyelvhasználó a *ballag ~ fut* szópár tagjainak a jelentését már távolabbinak véli egymástól, még ha a hasonlóság nagyságát, mértékét szavakkal nehezen is tudja megfogalmazni. Minden beszélő nyelvi kompetenciája azt sugallja, hogy ezek a szavak valamilyen módon egy csoportba tartoznak, hiszen mindegyik a

helyváltoztatással, mozgással kapcsolatos. Sőt belső nyelvi intuíciója alapján azt is érzi, hogy a *ballag, sétál* szavak mellé oda kívánkozik a *bandukol,* és ugyanúgy a *fut, szalad* szavak után felsorolható például a *rohan* szó. Mindeközben a nyelvi intuíció azt sugallja, hogy a *bandukol, sétál, ballag* és a *fut, szalad, rohan* két szóhármas valamiféle polaritást fejez ki, valamilyen képzeletbeli skála két végpontján helyezkedik el, és közéjük a *megy, jár, halad* szavak illenek be.

2. A szójelentés-hasonlóság mérésének eddigi útjai

Felvetődik a kérdés, hogy ezt az intuitív érzést, hogy az egyes szavak jelentése hol jobban, hol kevésbé hasonlít egymásra, ki tudjuk-e fejezni valamilyen számértékkel. Valamilyen módon meghatározható-e olyan mutató, amely kifejezi a szavak jelentésbeli hasonlóságát, illetve távolságát. Vagyis megállapítható-e, hogy a *ballag, sétál, bandukol, megy, jár, halad, fut, szalad, rohan* szavakból alkotott jelentéshasonlóság mátrixnak az egyes celláiban milyen mérőszámok helyezkednek el.

	ball	*sétá*	*ban*	*meg*	*iár*	*hal*	*fut*	*szal*	*roh*
ballag	1	?	?	?	?	?	?	?	?
sétál	-	1	?	?	?	?	?	?	?
banduk	-	-	1	?	?	?	?	?	?
megy	-	-	-	1	?	?	?	?	?
jár	-	-	-	-	1	?	?	?	?
halad	-	-	-	-	-	1	?	?	?
fut	-	-	-	-	-	-	1	?	?
szalad	-	-	-	-	-	-	-	1	?
rohan	-	-	-	-	-	-	-	-	1

A feladat tehát N szó (objektum) hasonlóságának kiszámításakor az (N * (N–1)) / 2 darab mérőszám meghatározása, majd továbblépésként a hasonlósági mátrix matematikai módszerekkel való feldolgozása, fogalomkörök automatikus előállítása.

Két szó jelentéshasonlósága meghatározásának 5 igen jól elkülöníthető útját ismerteti a szakirodalom:

1. A pszicholingvisztikai indíttatású ún. asszociációs tesztek és módszerek a legrégebbiek [Osgood 1952], [Osgood 1957], [Miller 1971]. Ezeknek a következő négy főbb típusa van: 1. skálázás, 2. asszociáció, 3. helyettesítés, 4. osztályozás. Az asszociációs fogalmi pszichológiai kísérletek tapasztalatinak ismertetése során helyesen mutat rá Varga Dénes, hogy az eredmény sok esetben különböző szintű, akár 8–10 féle asszociációból tevődik össze: 1. tárgy / tulajdonság, 2. közös / alárendeltség, 3. ellentét, 4. faj / nem, 5. hasonlóság, 6. objektum / cselekvés, 7. közös előfordulás, 8. rész /egész, 9. ok / okozat [Varga 1968]. Mi is úgy véljük, hogy az asszociációs tesztek éppen ezért finom jelentéshasonlóságok meghatározására alkalmatlanok, még ha erre jó néhány kísérlet történt is az utóbbi években.

2. Szójelentés-hasonlóságok meghatározására gyakran felhasználják a szóanyagukat hierarchikusan bemutató fogalomköri szótárakat. Ezek közül is a Roget's thesaurus [Kirkpatrick 1998] a legtöbbet vizsgált és alkalmazott [Morris, Hirst 1991], [Manuba, Takco 1994].

3. A fogalomköri egynyelvű szótárak mellett, az egynyelvű értelmező szótárak is számos vizsgálatnak jelentették a kiinduló pontját. Hideki és Teiki az LDOC (Longman Dictionary of Contemporary English) szótár definícióiból kiindulva végeztek szójelentés-hasonlósági vizsgálatokat. Rámutattak, hogy az egynyelvű értelmező szótárakból azért is célszerű kiindulni, mert azok értelmezéseit lexikográfusok szerkesztették meg értő módon [Hideki, Teiji 1993].

4. Minden kétséget kizárólag a WordNet megjelenése áttörést hozott a természetes nyelvek gépi feldolgozásában és ezen belül is a szópárok jelentéshasonlóságának vizsgálatában, Az új lehetőségekre G. A. Miller hívta fel talán először a figyelmet [Miller 1990], majd számos tudományos és népszerűsítő tanulmány, könyv ismerteti azokat [Fellbaum 1998]. A WordNet alapú szójelentés-hasonlóságok számításának 5 féle módját fejlesztették ki az elmúlt évtizedben [Leacock, Chodorow 1998], [Jiang, Conrath 1997], [Resnik 1995], [Lin 1998], [Hirst, St. Onge 1998], ezeket összefoglalóan ismerteti a Budanitsky, Hirst szerzőpáros [Budanitsky, Hirst 2001].

5. Korpusz alapú szójelentés-hasonlósági vizsgálatok igen sok helyen folynak, ezek alapja a ma már több nyelven is létező, nagyságrendileg 100 millió szövegszót tartalmazó nemzeti nyelvű korpuszok [Jiang, Conrath 1998].

3. Magyar szavak jelentéshasonlóságának megállapításához a magyar nyelvi tudásbázis kiválasztása

Ha magyar szavak jelentését, illetve szópárok jelentéshasonlóságát kívánjuk vizsgálni, először is ésszerűnek tűnik, hogy szótárhoz, esetünkben a legnagyobb magyar köznyelvi szótárhoz, *A Magyar Nyelv Értelmező Szótárához* [ÉrtSz.] forduljunk. A következő jelentésmeghatározásokat, definíciókat találjuk ott, példaképp a *ballag* és a *sétál* szócikkeket adjuk közre:

ballag: tn ige

1. <Ember, állat> lassan kényelmesen lépegetve megy vagy jön.

2. (Isk) <Végzős közép- vagy főiskolás diák> az utolsó tanév legvégén az intézet helyiségeit és környékét csoportosan, hagyományos módon bejárva búcsúzik az iskolától.

3. felé ballag: <személy> az ötvenedik, a hatvanadik, hetvenedik stb. évéhez közeledik.

sétál: tn ige

1. <Személy> testmozgás, levegőzés végett, vagy kedvtelésből lassú, nyugodt léptekkel megy, jár; sétát tesz.

|| **a.** <Rendszerint zárt térben> fel s alá, ide-oda jár, járkál.

|| **b.** (nép) <Ingaóra sétálója> ide-oda leng, <az óra> jár.

2. sétál valahová: sétálva valahova megy.

3. (rosszalló) <Személy> feltűnő ráérő(s)en, illetve a szükségesnél lassabban, kényelmesen lépked.

|| **a.** (rosszalló) Henyén, dologkerülő módon jár-kel.

4. (szoc. e. biz) Állás, munka nélkül van.

Amennyiben többé-kevésbé automatikusan, valamelyest algoritmizáltan szeretnénk az ÉrtSz.-nek a fenti jelentésmeghatározásaiból, azaz szöveges, mondat formájú definícióiból a jelentéshasonlóságokat jelző mérőszámokat meghatározni, nehéz dolgunk lenne. Ezért az ÉrtSz.-et, mint segédeszközt és kiindulási pontot a jelentések hasonlóságának meghatározásához el kell vetnünk.

Az ÉrtSz. tanulmányozása a fentiek ellenére nem volt hiábavaló, hiszen rávilágít arra, hogy egy-egy szónak a jelentése természetszerűleg aljelentésekből tevődik össze. Így, amikor szavak jelentésének hasonlóságát kívánjuk meghatározni, akkor a két vagy több összevetni kívánt szónak az aljelentéseit kell összevetnünk, összehasonlítanunk.

4. A *Magyar szókincstár* mint magyar szavak jelentéshasonlóság mérésének a nyelvi tudásbázisa

Ha figyelmesen szemléljük az ÉrtSz. fenti definícióit, észrevehetjük, hogy a szótár szerkesztői a mondatszerű definíciókat igen gyakran szinonimákkal toldották meg. A gazdag tartalmú, szókészletünk elemeit lineárisan feldolgozó *Magyar szókincstár – Rokon értelmű szavak, szólások és ellentétek szótára* [Kiss 1998] előszavában a főszerkesztő a következőket írja: „A Magyar Szókincstárnak azon kívül, hogy szinonimaszótárként forgatva valamely szó szinonimáit, sőt ellentéteit is kikereshetjük belőle, van egy további haszna is. Ugyanis a szótár rokon értelmű szavai a maguk módján magyarázzák, értelmezik azt a címszót, mely alá be vannak sorolva. Így a *Magyar szókincstár* bizonyos tekintetben értelmező szótári funkciót is betölthet, hiszen az olvasó számára egy kevésbé ismert, homályos jelentésű szónak a jelentését a szinonimák pontosíthatják, megvilágosíthatják." Az idézet elegendő indokot szolgáltat arra, hogy a fenti 9 tagú (*ballag, sétál, bandukol, megy, jár, halad, fut, szalad, rohan)* szósornak megvizsgáljuk a szinonima sorait a *Magyar szókincstárban*, abból a célból, hogy azok alkalmasak-e jelentéshasonlóságok számításának kiinduló pontjaként. Példaképpen megmutatjuk, hogy a *Magyar szókincstárban* a *ballag* és a *sétál* címszavak alatt a következő szinonimasorokat találjuk:

ballag (ige)
1• bandukol, baktat, mendegél, megy, lépked, kullog, cammog, andalog, battyog, slattyog (biz), sétál, poroszkál, kutyog (táj), ballagdál (táj), ballókál (táj), bandikál (táj), bandukál (táj)
2• [iskolától] búcsúzik
sétál (ige)
1• jár, ballag, megy, gyalogol, mozog, kimozdul, levegőzik, kirándul, fordul egyet, kerül egyet
2• sétálgat, sétafikál (pej), sétifikál, járkál, jár-kel, grasszál (pej), korzózik (biz), flangál (biz), flangíroz (rég), spacíroz (rég), promenál (rég), andalog, lötyög (biz), kószál, kódorog (pej), őgyeleg, lődörög, csatangol, cselleng, gévalyog (táj)

A fenti mintából látszik, a *Magyar szókincstár* szerkesztői is (az ÉrtSz.-hez hasonlóan) egy-egy címszó jelentését aljelentésekre bontották, és ezeknek az aljelentéseknek adták meg a szinonimáit. A vizsgált 9 szónak összesen 48 aljelentése van (*ballag* 2, *sétál* 2, *bandukol* 1, *megy* 10, *jár* 13, *halad* 4, *fut* 9, *szalad* 5, *rohan* 2).

Ez azt jelenti, hogy a fenti 9 szó jelentéshasonlóságának meghatározáshoz egy 48 x 48 méretű hasonlósági mátrix kitöltése a feladat.

Míg az ÉrtSz. mondatszerű, szöveges definícióit nem tudtuk eredményesen felhasználni jelentéshasonlóságok meghatározására, úgy tűnik, hogy a *Magyar szókincstárban* közölt szinonimasorok jó kiinduló pontként szolgálhatnak a fenti feladat megoldásához.

A nyelvi tudásbázisként felhasználni kívánt *Magyar szókincstár* a következő jellemzőkkel rendelkezik:

szófaj	szótárban használt rövidítés	címszavak száma	jelentések száma	római sz. homonima	arab sz. homonima
főnév	fn	12400	19118	525	275
ige	ige	7618	15162	23	136
melléknév	mn	4491	6997	523	60
határozószó	hsz	932	1224	98	17
névmás	nm	105	156	18	6
kötőszó	ksz	71	98	37	14
mutatószó	msz	57	66	25	5
névutó	nu	53	85	54	9
számnév	szn	45	53	41	12
indulatszó	isz	13	15	60	9
igenév	ign	2	2	2	2
összesen		**25787**	**42976**	**1406**	**545**

5.1. A szójelentés-hasonlóság mérőszámának számítása

Egy-egy szinonimasort matematikai értelemben vett halmaznak tekinthetünk, amelyben az elemeket az egyes szinonimák adják. Egy szópár (pontosabban a szópár egy-egy aljelentésének) jelentéshasonlóság mérőszámának – melyet H-val jelölünk, H(SZÓa, SZÓb) – a kiszámítása a két halmaz hasonlóságának a meghatározását jelenti.

Például a *fut ~ szalad* szavak **első** aljelentéseinek a jelentéshasonlóságának a meghatározása a következő két halmaz hasonlóságának a kiszámításából áll:

fut 1 = fut, szalad, száguld, vágtat, üget, vágtázik, kocog, limel, rohan, robog, lohol, lót, nyargal, galoppozik, inal, iramlik, iramodik, cikázik, viharzik, kotor, darizik, sprintel, skerál, spurizik, teker, tép, kóstat

szalad 1 = szalad, fut, siet, repül, futkos, fáradozik, spurizik, rohan, lohol, lót-fut, szedi a lábát, iramlik, kotrecel, limel, lófol, inal, őringel, rőföl, trappol, skerál

Két halmaz hasonlósága mérésének kiinduló pontja a két halmazban előforduló közös elemek (P_{11}) és csak az egyes halmazokban meglévő elemek számának (P_{10}, P_{01}) valamiféle összevetése. Párniczky Gábor 1976-ban megjelent művében halmazok hasonlóságának meghatározására öt képletet ismertet (Russel és Rao, Sokal és Michene, Jaccard, Yule, Csuprov) [Párniczky1976]. Számunkra a Jaccard-képlet alkalmazása tűnik célszerűnek. Két szóhalmaz (két szinonimasor) metszete azokból a szavakból áll, amelyek közösek a két halmazban, két halmaz uniója az összes szó

(azaz az összes szinonima) együttese. Így a Jaccard képlettel eredményként kapott metszet – unió arány, azaz a hasonlósági mérték (H), 0 és 1 közé esik. Képletben:

$$H = P_{11} / (P_{11} + P_{10} + P_{01}). \quad \text{(Jaccard)} \qquad (1)$$

A metszet – unió arány használatának hatásos nyelvészeti alkalmazását a szakirodalom is alátámasztja. Példaképpen a *fut ~ szalad* hasonlóságának számítása:

P_{11} (közös elemek) = 9 szó: *fut, inal, iramlik, limel, lohol, rohan, skerál, spurizik, szalad.*

P_{10} (a *fut*-ban meglévő és a *szalad*-ban nem előforduló elemek) =18 *cikázik, darizik, galoppozik, iramodik, kocog, kóstat, kotor, lót, nyargal, robog, sprintel, száguld, teker, tép, üget, vágtat, vágtázik, viharzik.*

P_{01} (a *szalad*-ban meglévő és a *fut*-ban nem előforduló elemek) = 11 szó: *fáradozik, futkos, kotrecel, lófol, lót-fut, őringel, repül, rőföl, siet, szedi a lábát, trappol.*

		fut szinonimasora	
		1	0
szalad szinonimasora	1	9	11
	0	18	

$$H(\textit{fut, szalad}) = P_{11} / (P_{11} + P_{10} + P_{01}) = 9 / (9 + 11 + 18) \approx 0.24. \qquad (2)$$

Számításaink szerint tehát a *fut ~ szalad* szavak **első** aljelentéseinek a hasonlósága a Jaccard-képlet alapján (súlyozás nélkül) számolva: 0.24.

5.2. A szójelentés-hasonlóság mérőszámának tulajdonságai

A szójelentés-hasonlóság mérőszámára (H) a következő tulajdonságok teljesülnek:

– A két szó hasonlóságát mutató mérőszám 0 és 1 közé esik:
$0 \leq H(\text{SZÓa}, \text{SZÓb}) \leq 1$;

– A szópárok jelentéshasonlósága szimmetrikus, azaz a hasonlóság nem függ a szavak sorrendjétől (ezért a hasonlósági mátrix szimmetrikus és elegendő csak az egyik felét, pl. a főátló feletti részét megfelelő módon kitöltenünk):
$H(\text{SZÓa}, \text{SZÓb}) = H(\text{SZÓb}, \text{SZÓa})$;

– Minden szó önmagával vett hasonlósága 1, ezt jelzik a hasonlósági mátrix főátlójában szereplő 1 értékek:
$H(\text{SZÓa}, \text{SZÓa}) = 1$.

Akkor mondjuk, hogy SZÓb jobban hasonlít SZÓc-re mint a SZÓa-ra, ha
$H(\text{SZÓa}, \text{SZÓb}) \leq H(\text{SZÓc}, \text{SZÓb})$.

5.3. Az elemek súlyozásának szükségessége

Párniczky Gábor [Párniczky 1976] könyve felhívja a figyelmünket arra, hogy a gyakorlatban sok esetben a vizsgált halmazok egyes elemei nem egyforma fontosak, ezért célszerű, akár szubjektív módon is – súlyok meghatározása. Mi úgy döntöttünk,

figyelembe véve, hogy a vizsgált szinonimasorokban elhelyezkedő szavak „fontossága" a címszótól távolodva csökken, hogy a szinonimasor egyes tagjait olyan szorzóval látjuk el, amely a fenti szemléletet tükrözi. Ezt indokolja a *Magyar szókincstár* már idézett előszavában olvasható szerkesztői szándék is: „A szerkesztők a szinonimasorokban helyet foglaló adatokat úgy rendezték, hogy a címszó jelentéséhez közelebb álló szinonimák a sor elején álljanak, míg a címszó jelentésétől távolabb eső szavak a sor vége felé helyezkedjenek el." Ezért egy n elemű szinonimasor i-dik tagja kísérletünkben a következő szórzósúlyt kapja:

$$W_i = (n + 1 - i) / n$$

A fenti (1)-es képletben tehát a P_{11}, a P_{10} és a P_{01} a megfelelő elemek súlyainak összegét jelenti.

A *fut ~ szalad* szópárok első aljelentésének jelentéshasonlósága a fenti módon meghatározott súlyokkal számolva:

H_w (*fut, szalad*) = $P_{11} / (P_{11} + P_{10} + P_{01}) = ((5+5.29)/2) / (((5+5.29)/2) + 8.04 + 4.92) \approx 0.28$.

6. Fogalomkörök létrehozása automatikus osztályozással

Az automatikus osztályozás olyan eljárás, ami csoportok – jelen kísérletünkben fogalomkörök – képzését hivatott elősegíteni. Esetünkben a szófajokra szétbontott jelentésmátrixok alkalmasak arra, hogy az automatikus osztályozás valamely módszerével fogalomköröket hozzunk létre. Ez annál is inkább sürgetően időszerű a magyar nyelv esetében, mert míg a magyar állandósult szókapcsolatoknak létezik korszerű fogalomköri feldolgozása [Bárdosi 2003], addig a magyar szavaknak nincs ilyen jellegű csoportosítása.

A hagyományos osztályozás jellegzetességei, melyet mi is megkövetelünk nyelvi anyagunk feldolgozásakor:

1. Átfedés mentesség: az osztályozandó halmaz elemei, csak egy osztályban szerepelhetnek;
2. Teljesség: minden elem beletartozik valamelyik osztályba;
3. Homogenitás: az egymáshoz hasonló egységek lehetőség szerint egy adott osztályba kerüljenek.

Az automatikus osztályozás végrehajtása két lépésben történik.

Először páronként vizsgáljuk meg a halmaz elemeit és ennek eredményeként:

– hasonlósági értéket számítunk ki (ezt tettük mi is), vagy

– adott relációhoz tartozó párokat jelölünk ki.

Második lépésben ezt követően valamely csoportképző algoritmus segítségével az egymáshoz közeli elemeket osztályokba rendezzük.

Esetünkben lényeges kiemelni, hogy adott szófajhoz tartozó címszónak csak adott szófajú szinonimái lehetnek. Így a jelentések számából kiindulva a 42976 x 42946 nagyságú jelentésmátrix kisebb, lényegében 3, a főnevek, az igék és a melléknevek mátrixaira bontható. A főnevek jelentésmátrixa 19118 x 19118, az igéké 15162 x 15162 és a melléknevéké 6997 x 6997 méretű. Az eredeti jelentésmátrix ezen dekomponálása az adatok feldolgozását megkönnyíti.

Így a szófajok szerint szétbontott jelentéshasonlósági mátrixok megalkotása után megfelelő módszert kell találnunk az adat mátrix mögött meghúzódó szerkezet, lényegi nyelvi tartalom felderítésére, azaz hatékony eljárást kell keresnünk az automatikus osztályozás elvégzése, a fogalomkörök generálása. A rendelkezésre álló eszközök közül a **szinguláris érték dekompozíciót** (SVD) választottuk [Kennedy1980], amivel faktorokat, illetve osztályokat is meg lehet határozni. Az SVD egyik jó tulajdonsága az, hogy nem csak négyzetes jelentésmátrixok esetén alkalmazható, hanem téglalap alakúak estén is (pl. jelentés részmátrix), ami finomabb vizsgálatok elvégzését teszi lehetővé. Az SVD differenciálgeometriai vizsgálatával foglalkozik [Rapcsák 2004].

A fent már ismertetett 48x48 méretű jelentésmátrixra elvégeztük az SVD-t és azt kaptuk, hogy a mátrix rangja 48. Tehát a vizsgált szavak jelentését, (természetszerűleg az aljelentéseket) az őket követő szinonimasorok erősen jellemzik. Az egyes szavaknak a jelentésmátrixban elfoglalt súlyát mutatják az SVD során kiszámolt szinguláris értékek:

A szinguláris értékek					
ballag1		jár2		fut1	
ballag2		jár3		fut2	
sétál1		jár4		fut3	
sétál2		jár5		fut4	
bandu-		jár6		fut5	
megy1		jár7		fut6	
megy2		jár8		fut7	
megy3		jár9		fut8	
megy4		jár10		fut9	
megy5		jár11		szalad1	
megy6		jár12		szalad2	
megy7		jár13		szalad3	
megy8		halad1		szalad4	
megy9		halad2		szalad5	
megy10		halad3		rohan1	
jár1		halad4		rohan2	

A 48 x 48 méretű jelentésmátrixban rejlő hasonlósági kapcsolatokat és ezek mértékét egy grafikus ún. dendogram alakzattal szemléltethetjük. Közreadjuk a fentiekben vizsgált 9 ige, 48 aljelentésének hasonlóságát mutató dendogramot. Az ábrán például jól megfigyelhető, hogy a „gép"-re vonatkozó *jár* 4. aljelentése és a *megy* 5. aljelentése hasonló. E természetes is, hiszen a *Magyar szókincstárban* ezek mellett a következő szinonima sorok állnak:

jár 4 = működik, üzemel, dolgozik, forog, köröz, kering, cirkulál, funkcionál, szuperál
megy 5 = jár, működik, dolgozik, közlekedik, üzemel

7. Eredményeink

Úgy véljük, kísérletünk sikeres, a magyar szavaknak, pontosabban a szavak aljelentéseinek hasonlóságát egzakt módon, eredményesen határozhatjuk meg a *Magyar szókincstárban* található szinonimasorokból kiindulva. Rámutattunk, arra hogy

ha a jelentéshasonlósági mérőszámokat mátrixban helyezzük el, ez a mátrix alkalmas kiindulási alap jelentéscsoportok, fogalomkörök automatikus képzésére.

A közreadott dendogramot szemlélve, megállapíthatjuk, hogy az anyanyelvi beszélő nyelvi kompetenciájával egyező számítási eredmény született automatikus módon.

Meggyőződésünk, hogy a módszerünk nyelvfüggetlen, a magyaron kívül más nyelvre is átvihető. Eredményesen alkalmazható nyelvi tudásbázisként az adott nyelv szinonima szótára, és hatásosan használható fel szójelentés-hasolóságok mérésére, fogalomkörök generálására.

Melléklet: A 9 szó, 48 aljelentéséből számított szójelentés-hasonlóság dendogramos ábrázolása.

Bibliográfia

[Bárdosi 2003] BÁRDOSI VILMOS (főszerk.): Magyar szólástár. Szólások, helyzetmondatok, közmondások értelmező és fogalomköri szótára. TINTA Könyvkiadó, 2003.

[ÉrtSz.] BÁRCZI GÉZA és ORSZÁGH LÁSZLÓ (főszerkesztők): A Magyar Nyelv Értelmező Szótára I–VII. Akadémiai Kiadó, Budapest, 1959–1961.

[Fellbaum 1998] Christiane Fellbaum (ed): WordNet: An Electronic Lexical Database. Cambridge, MIT Press, 1998.

[Hideki, Teiji 1993] HIDEKI KOZIMA and TEIJI FURUGORI: Similarity Between Words Computed by Spreading Activation on an English Dictionary. Proceedings of EACL-93. 232–239, 1993.

[Hirst 1998] HIRST G. and ST. ONGE D.: Lexical Chains as representations of context for the detection and correction of malapropisms. In Fellbaum 305–332. 1998.

[Jiang, Conrath 1998] J. JIANG and D. W. CONRATH: Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. Proceedings of the 10th International Conference: Research on Computational Linguistics (ROCLING X), 19–33, 1998.

[Kennedy 1980] W. J. KENNEDY and J. E. GENTLE: Statistical computing, Marcel Dekker, New York, Basel, 1980.

[Kiefer 2000] KIEFER FERENC: Jelentéselmélet. Budapest, 2000.

[Kirkpatrick 1998] BETTY KIRKPATRCK: Roget's Thesaurus of English Words and Phrases. Harmondsworth, Middlesex, Penguin, 1998.

[Kiss 1998] KISS GÁBOR (főszerkesztő): Magyar szókincstár. Rokon értelmű szavak, szólások és ellentétek szótára. TINTA Könyvkiadó, Budapest, 1998\1

[Kiss, Kiss 2004] KISS GÁBOR és KISS MÁRTON: Kísérlet egy szócsoport elemei jelentéshasonlóságának meghatározására. In.: „...még onnét is eljutni a túlra..." Nyelvészeti és irodalmi tanulmányok Horváth Katalin tiszteletére. 159–165. oldal. TINTA Könyvkiadó, 2004.

[LDOC 1987] Longman Dictionary of Contemporary English. Longman, Harlow, Essex, new edition, 1987.

[Leacock, Chodorow 1998] LEACOCK C. and CHODOROW M.: 1998. Combining local context and WordNet similarity for word sense identification. In Fellbaum 265–283. 1998.

[Lin 1998] LIN D.: An information-theoretic definition of similarity. In Proceedings of the 15th International Conference on Machine Learning, Madison, WI.

[Manabu, Takco 1994] MANABU OKUMURA and TAKO HONDA: Word sense Disambiguation and Text Segmentation Based on Lexical Cohesion. In Proceedings of COLINGS-94. Vol. 2, 755–761, 1994.

[Miller 1990] MILLER, G. A.: WordNet: An on-line lexical Database. International Journal of Lexicography, 3/4, Special Issue, 235–312. 1990.

[Miler 1971] GEORGE A. MILLER: Empirikus módszerek a szemantika kutatásában. Fordította: Siklay István. In: Pszicholingvisztika és kommunikációkutatás. Szöveggyűjtemény. Válogatta ás a bevezetőt írta: Pléh Csaba. Tömegkommunikációs Kutatóközpont, Budapest, 1977. (Empirical methods in the Study of Semantics. In.: Danny D. Steiberg és Leon A. Jakobovits (szerk.): Semantics: An interdisciplinary reader in philosphy, linguistics and psychology. London, Cambridge University Press, 1971, 569–585.)

[Morris, Hils 1991] J. MORRIS and G. HIRST: Lexical cohesion computed by thesaural relations as an indicator of the structure of text. Computational Linguistics, 17., 21–48, 1991.

[Martinkó 2001] MARTINKÓ ANDRÁS: A szó jelentése. Lazi Könyvkiadó, Szeged, 2001.

[Osgood 1952] C. E. OSGOOD: The natural and measurement of meaning. Psychological Bulletin, 49, 197–237, 1952

[Osgood, Succi, Tannenbaum 1957] C. E. OSGOOD and G. J. SUCCI and P. H. TANNENBAUM: The Measurement of Meaning. University of Illinois Press, Urbana, 1957

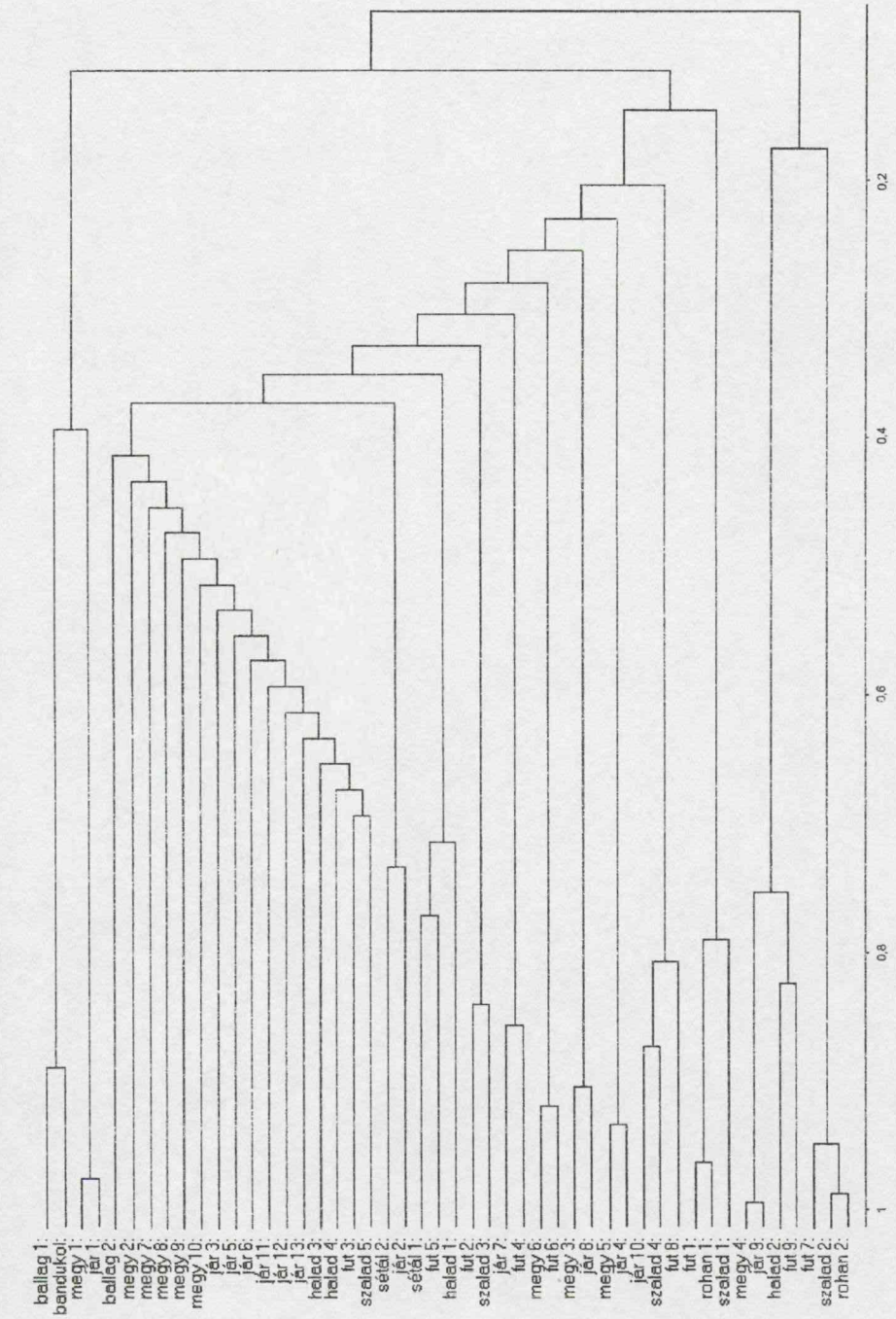
[Párniczky 1976] PÁRNICZKY VIKTOR: A statisztika alapjai. Statisztikai Kiadó Vállalat, A korszerű informatika könyvtára 8. 1976.

[Rapcsák 2004] RAPCSÁK TAMÁS: Some optimization problems in multivariate statistics, *Journal of Global Optimization* 28 (2004) 217–228.

[Resnik 1995] RESNIK P: Using information content to evaluate semantic similarity. In Proceedings of the 14th International Joint Conference on Artificial Intelligence, pages 448–453, Montreal. 1995.

[Varga 1969] VARGA DÉNES: Információs tezauruszok készítésének módszertana. Országos Műszaki Könyvtár és Dokumentációs Központ. Budapest, 1969.

[Villó, Kiss 1996] VILLÓ ILDIKÓ és KISS GÁBOR: Mozgást jelentő igék szinonimitásának vizsgálata. In.: Emlékkönyv B. Lőrinczy Éva hetvenedik születésnapjára. Szerk.: Bánki Judit. 123–128. oldal. MTA Nyelvtudományi Intézete, Budapest, 1996


A ballag, sétál, bandukol, megy, jár, halad, fut, szalad, rohan szavak 42 aljelentésének hasonlósági ábrázolása
ballag 1.
bandukol.
megy 1.
jár 1.
ballag 2.
megy 2.
megy 7.
megy 8.
megy 9.
megy 10.
jár 3.
jár 5.
jár 6.
jár 11.
jár 12.
jár 13.
halad 3.
halad 4.
fut 3.
szalad 5.
sétál 2.
jár 2.
sétál 1.
fut 5.
halad 1.
fut 2.
szalad 3.
jár 7.
fut 4.
megy 6.
fut 6.
megy 3.
jár 8.
megy 5.
jár 4.
jár 10.
szalad 4.
fut 8.
fut 1.
rohan 1.
szalad 1.
megy 4.
jár 9.
halad 2.
fut 9.
fut 7.
szalad 2.
rohan 2.
1
0,8
0,6
0,4
0,2

II. Kivonatolás

Programcsomag információkinyerési kutatások támogatására

Alexin Zoltán[1], Gyimóthy Tibor[1], Csirik János[1]

Szegedi Tudományegyetem, TTK, Informatikai Tanszékcsoport,
Szeged Árpád tér 2., e-mail:{alexin,gyimothy,csirik}@inf.u-szeged.hu

Kivonat A publikációban bemutatásra kerül egy információkinyerési kutatásokat támogató programcsomag, amelynek moduljai a nyers szöveg beolvasásától kezdve a végeredmény webes megjelenítéséig minden szükséges funkciót megvalósítanak. A modulok egymással szabványos TEI XML állományok segítségével kommunikálnak, amelyek a feldolgozás tetszőleges szakaszában elemezhetők. A technológia ezen a módon támogatást nyújt az egyes modulok önálló fejlesztéséhez és teszteléséhez. A fontosabb modulok: a szegmentáló, morfológiai elemző, szófaji egyértelműsítő, felszíni szintaktikai elemző, szemantikai bővítménykezelő, eseménymintákat felismerő mintaillesztő és webes megjelenítő modul. A szerzők a programcsomag működését egy kísérleti rendszeren mutatják be, amely üzleti rövidhírekből gyűjt különböző információkat. [1]

Kulcsszavak: információkinyerés, természetesnyelv-feldolgozás, felszíni szintaktikai elemzés

1. Bevezetés

Az információkinyerés (IE, Information Extraction) technológiájának kutatása dinamikusan fejlődő terület a természetesnyelv-feldolgozásban. Az Interneten megjelenő hatalmas információtömeg gépi feldolgozása és a kívánt információ tömör formában történő összegyűjtése napi szükséglet, amelyre a gazdaság, a tudomány, a politika, de akár a hírszerzés területén is van igény. Míg az információ visszakeresés (IR, Information Retrieval), amely a webes kereső programok jellemző tevékenysége, arra irányul, hogy a felhasználó igényeinek megfelelő dokumentumokat változatlan formában bocsássa rendelkezésre, addig az információkinyerés célja a megtalált dokumentumokban a lényeges információ megjelölése, majd összegyűjtése. A számítógéppel támogatott szövegtömörítés, kivonatolás és az információkinyerés szoros kapcsolatban áll egymással.

Az információkinyeréssel foglalkozó rendszerek nem törekednek a szövegek teljes megértésére és analízisére, a fő követelmény velük szemben a nagy kapacitás, a gyorsaság és egy elfogadható szintű pontosság. Rendszerint megelégednek

[1] A szerzők köszönetüket fejezik ki az Oktatási Minisztériumnak, amely az NKFP 2/017/2001 projekt keretében az itt ismertetésre kerülő kutatást támogatta.

a mondatok fontosabb szereplőinek azonosításával anélkül, hogy részletes szintaktikai elemzést végeznének. Ehhez egy ún. felszíni elemzés (shallow parsing) végrehajtására van szükség. A szereplők és mondatbeli szerepeik azonosításában fontos szerepet játszanak a tulajdonnevek (*named entityk*). A gyakori közéleti szereplők, cégek, európai és magyar városok nevének felismerése egy nagyméretű lexikon alapján, a szófaji elemzéstől függetlenül történik.

A továbbiakban bemutatásra kerül egy, a Szegedi Tudományegyetem Informatikai Tanszékcsoportjában kifejlesztett programcsomag, amely az információkinyerési kutatások támogatására készült. A koncepció egyik legfontosabb szempontja a modularitás volt, így a komponensek egymástól függetlenül fejleszthetők. A modulok egymástól függetlenül futtathatók, így a feldolgozás minden egyes lépése nyomon követhető és ellenőrizhető. A fenti tulajdonságok nagy súllyal esnek latba a kutatások kezdeti szakaszában, amikor az egyes modulok kísérleti fejlesztése folyik. Az egyes modulok szabványos kommunikációja megkönnyíti a modulok gyors cseréjét, a leghatékonyabb mód- szer kiválasztását.

Az elkészült rendszert gazdasági témájú rövidhírek feldolgozására alkalmazták. Az MTI-Eco, Business+ szolgáltatását [2] felhasználva egy 6453 hírből álló adatbázist hoztak létre, amely egyaránt szolgált tréning- és tesztelési célokat. A kutatási és fejlesztési munkában, az újabb, egyre tökéletesebb modulok készítése során egy hasonló rendszer nem nélkülözhető. A jelenlegi fejlesztések a kulcspozíciót betöltő modulokra irányulnak: a szóalaktani egyértelműsítő, a felszíni szintaktikai elemző és az eseménymintákat, ún. *szemantikus kereteket* felismerő modulra.

2. A programcsomag fontosabb tulajdonságai

Az ismertetésre kerülő programcsomag moduljai végigkövetik a nyers szöveg feldolgozásának egyes lépéseit a szöveg mondatokra és szavakra bontásától kezdve, a szóalaktani elemzésen, egyértelműsítésen, és a felszíni elemzésen át, a mondatminták felismeréséig, a szereplők azonosításáig, majd pedig az eredmények tömörített formában történő webes megjelenítéséig.

2.1. Az információkinyeréshez kapcsolódó modulok fejlesztését támogató adatbázis: a Szeged Korpusz 2.0

Az Oktatási Minisztérum által támogatott IKTA 27/2000 projekt keretében készült el a Szeged Korpusz 1.0-s változata[1], amely egy szófajilag elemzett, majd kézzel egyértelműsített adatbázis volt. Ezt az információkinyerési kutatások támogatására az MTA Nyelvtudományi Intézettel és a MorphoLogic Kft.-vel közös konzorcium jelentősen továbbfejlesztette. A Szeged Korpusz újabb változata [3] hat különböző témakörben gyűjtött, összesen 1,2 millió szót tartalmazó, számítógéppel feldolgozható szöveg. Ennek az állománynak egy mintegy 200 ezer

[2] Magyar Távírati Iroda, `http://www.mti.hu`

[3] SZTE, Informatikai Tanszékcsoport, Nyelvtechnológiai Csoport: `http://www.inf.u-szeged.hu/hlt`

szavas részét képezi a bevezetőben már említett 6453 MTI rövidhírt tartalmazó anyag.

A Szeged Korpusz 2.0[5][6] kialakítása során a készítők figyelembe vették egy majdani információkinyeréssel kapcsolatos alkalmazás fejlesztésekor felmerülő igényeket. Elsősorban a felszíni elemzés jelenségeinek tanulmányozása érdekében került sor a szövegben a főnévi szerkezetek teljes annotációjára, azaz minden egyes főnévi szerkezetet és annak belső is szerkezetét megjelölték. A kezdeti annotációt számítógépes program állította elő, amelyet azután nyelvész szakértők egyenként ellenőriztek, és hiba esetén javítottak. A szövegadatbázis nagy mennyiségben tartalmaz tulajdonneveket, a hírekben gyakran előforduló cégek és személyiségek neveit.

A korpusz rövidhíreit a fentieken kívül további információkkal is kiegészítették, nevezetesen az IPTC[4] által javasolt tematikus kódolással. A tematikus kódok alapján a gazdasági rövidhírek egy nagyon finom osztályozását kapták: csupán az üzleti élet témakörén belül 41 alkategóriát különböztettek meg például: könyvvizsgálás/auditálás, vállalati közgyűlés, éves beszámoló, adás-vétel stb. A 6453 rövidhírből 2478 hír kapcsolódott a vállalati üzleti élethez, a többi kereskedelmi, tőzsdei, pénzügyi és egyéb hír volt. A főnevekhez kapcsolódó lexikális ismeretek tárolására egy egyszerű felépítésű ontológiai adatbázist hoztak létre a konzorciumi partnerek. Ebből megtudható például, hogy egy adott főnév élőlény, élettelen tárgy, vagy pénznem-e, illetve tulajdonnévként személy, cég vagy bank neve-e. Az eseménymintákat felismerő modul nagy mértékben támaszkodik ezekre az információkra. Az ontológiai adatbázis jelentős fejlesztése folyamatban van.

2.2. A modulok szabványos XML felületen történő kommunikációja

A Szeged Korpusz 2.0 készítésekor a TEI [5] XML[6] dokumentumtárolási szabványt vették alapul. Ez az elektronikus szövegek tárolására szolgáló technológia széles körben elterjedt. Az Informatikai Tanszékcsoport a TEI-konzorcium alapító tagja, a TEI konzorcium honlapján 119 különböző projekt ismertetője található a világ minden tájáról, beleértve a Szeged Korpuszét is. Az XML-formátum lehetővé teszi, hogy a nyers szöveghez ún. metainformációkat rendeljenek, amelyeket címkék jeleznek a szövegben. Az 1. ábrán egy példa található erre: a `<sentence>` címke a mondatokat, a `<word>` a szavakat, míg az `<mscat>` címke a morfo-szintaktikai kategóriát jelzi. Általában a címkék opcionálisak, ezért ugyanahhoz a szöveghez a metainformációk egyre bővülő halmazát lehet hozzárendelni a feldolgozás előrehaladtával.

A természetes nyelvi feldolgozás, az ismertetett programcsomag moduljainak eredményei a nyers szövegbe metainformációként kerülnek be, például az alaktani elemzés eredménye, vagy a felszíni elemzés eredménye. Minden egyes modul növeli a kiindulási szöveg metainformáció-tartalmát. A TEI-szabvány a szükséges

[4] Az IPTC (International Press Telecommunication Council) által javasolt tematikus kódolás: `http://www.iptc.org/site/subject-codes/subjectcode.html`

[5] Text Encoding Initiative, `http://www.tei-c.org`

[6] XML információs oldal: `http://www.xml.org`

címkék leírásával, a metainformációk belső struktúrájának formalizálásával nyújt segítséget ehhez a feladathoz.

```
<xml>
  <sentence id="1.1">
    <word>A<mscat>NE</mscat></word>
    <word>kutya<mscat>FN</mscat></word>
    <word>ugat<mscat>IGE</mscat></word>
    <punctuation>.</punctuation>
  </sentence>
</xml>
```

1. ábra. Egy XML állomány részlete

3. A programcsomag fontosabb moduljai

A következőkben a programcsomag egyes moduljai kerülnek részletesebben is bemutatásra. A szerzők által vezetett kutatócsoport egyik fő célkitűzése az volt, hogy a modulok fejlesztéséhez lehetőség szerint gépi tanuló algoritmusokat alkalmazzanak. A tanuló rendszerek segítségével és a nyelvészeti szakértők tudásának ötvözésével nemcsak az általános jellegzetességek, hanem a tréningadatok feldolgozásával kapott speiális nyelvészeti jegyek is kezelhetők, ezáltal a programok hatékonysága nagyobb lehet. Növelve a tréningadatok méretét a program pontossága nőhet, cserélve a tréning adatokat a modulok speciális területekre hangolhatók.

3.1. A beolvasott szöveg szegmentálását végző modul

A beolvasott szöveg XML adatbázissá alakítása és az alapvető metainformációk (fejezet-, bekezdés-, mondat-, szóstruktúra) meghatározása a feldolgozás első lépcsője[2] [11]. A természetes nyelvi szövegekben számos különböző fajta szó jellegű, de a szótárakban nem szereplő lexikai elem található (szám, dátum, gépkocsirendszám, e-mail cím, stb.), amelyek felismerésére valamint a mondathatárok megállapítására egy formális-nyelvi eszközöket alkalmazó modul készült. A modul a GNU Flex [7] reguláris automatagenerátor eszközt használja. Ebben reguláris kifejezések írják le az ún. tokeneket (2. ábra). A *flex* program a reguláris kifejezésekből C programot készít, amely a szegmentáló modul magját alkotja. A mondatokra és a szavakra bontás hatásfoka igen jó, a hibásan felismert tokenek aránya nem több, mint 0,5%.

3.2. A szófaji elemző és egyértelműsítő modul

A programcsomagban több, különböző elven működő szófaji egyértelműsítő (*part-of-speech tagger*) modul található. Alapvetően minden egyes modul gépi tanuló algoritmussal meghatározott egyértelműsítési szabályokkal dolgozik – a

[7] A GNU Flex honlapja: `http://www.gnu.org/software/flex`

```
/* ponttal tagolt számok, pl. 12.000 */
NUMDOT [0-9]{1,3}("."[0-9]{3})+
NUMDIGIT ([0-9]+",")?[0-9]+
```

2. ábra. Reguláris kifejezések a Flex definíciós fájljában

különbség a felhasznált tanuló algoritmusokban van. A szabályok a többértelmű szó környezetében – előtte vagy utána – található más szavak morfo-szintaktikai kódjai, szótövei alapján hoznak döntést. Az egyik modul E. Brill TBL (Transformation Based Learning)[4] módszerén alapul, a második egy HMM (Hidden Markov Model) alapú algoritmus, amely a TnT tanuló módszert használja[3], a harmadik pedig egy logikai döntési szabályokat tanuló algoritmus[10], amelyet a tanszékcsoportban fejlesztettek ki. Fontosnak tartjuk megjegyezni, hogy bár az egyes tanuló módszerek forráskódjai nem állnak rendelkezésünkre, azonban a megtanult szabályokat végrehajtani képes programokból mindhárom esetben van saját forráskódú verzió is.

A modulok pontossága nagyon jónak mondható: az összes szóra vetített pontosság (*per-word accuracy*) 95,79% és 97,83% között mozgott[9]. Az információkinyerési alkalmazásokba bármelyik modul beépíthető, a modulok egymással teljes mértékben kompatibilisek. [8]

3.3. A felszíni szintaktikai elemző modul

A felszíni elemző feladata a mondatban szereplő főnévi struktúrák azonosítása. Ez a modul nem alkalmas egyéb bonyolultabb nyelvi szerkezetek, például határozói, jelzői, vagy igei szerkezetek felismerésére. Az elemző modul szintaktikai szabályait a Hócza András által készített RGLearn[8] algoritmus állítja elő. Az RGLearn a Szeged Korpusz 2.0-ból válogatott tréningadatok alapján Chomsky-féle formális szintaktikai szabályokat tanul.

A tanuló algoritmus a tréningadatokból gyűjtött főnéviszerkezet-fákból indul ki. Hasonló fákat keres, majd megpróbálja azokat egyesíteni. További eleme az algoritmusnak az ismétlődő, azonos morfo-szintaktikai kóddal rendelkező szavak sorozatának észlelése és egy általános rekurzív szabállyal történő helyettesítése. Azokat a régi szabályokat, amelyeket az újonnan megtanult általános szabályok magukba foglalnak, törli, és a folyamat addig folytatódik, amíg csak van lehetséges általánosítás.

A főnévi szerkezeteket felismerő modul a teljes Szeged Korpusz 2.0-n egy véletlen algoritmussal választott tréning- és attól független tesztadatok esetén 75,72% pontosságot ért el 81,69% találati arány mellett. Az üzleti rövidhírek esetén a pontosság 79,86% volt 86,63% találati aránnyal[8].

[8] A Szeged Korpusz hivatalosan az MSD szófaji kódolást használja, azonban a kutatók dolgoznak a Humor-kódolású verzión is. A tanszékcsoportban rendelkezésre áll kísérleti jelleggel mindhárom POS-tagger Humor-kódokkal tanított verziója is.

4. Az információkinyerés modellje és technológiája

Információkinyerő rendszerek megvalósításakor igyekeznek elkerülni a bonyolult szintaktikai szerkezetek azonosítását, mert így nagy mértékben gyorsítható a program működése. Az NKFP projekt résztvevői a feladatot mintaillesztési feladattá alakították át az alábbiakban ismertetésre kerülő módon.

4.1. Szemantikus keretek és felismerésük

A mondatok és a bennük leírt esemény absztrakt leírására bevezették a szemantikus keret (*semantic frame*) fogalmát. A szemantikus keret egy absztrakt eseménynek tekinthető, amely az írott szövegekben számos különböző megfogalmazásban (mondatban) fordulhat elő. A szemantikus keret tartalmaz egy fő cselekvést és ahhoz kapcsolódóan különböző szereplőket. A szereplők azonosítása lexikális, morfo-szintaktikai, felszíni elemzési és ontológiai attribútumaikra vonatkozó feltételekkel történik. Egy mondat akkor illeszthető egy szemantikus keretre, ha a keretben definiált fő ige és a szereplők a mondatbeliekkel mind egyenként azonosíthatók. A modul az előre kidolgozott szemantikus kereteket próbálja végig minden egyes mondaton. A szemantikus keretek további építőeleme az információs ablak (*information slot*). Amennyiben a keret illesztése sikeres, úgy az ablakban jelenik meg a keresett információ, amelyet a felhasználó keres.

A kezdeti kutatásokban a szemantikus keretek egyetlen mondatnak voltak megfeleltethetők, az információs ablakok pedig a szereplőkkel voltak azonosak. A rendszer továbbfejlesztése folyamatban van elsősorban a több mondattal megfogalmazott események egyetlen kerettel történő leírásának irányában[7].

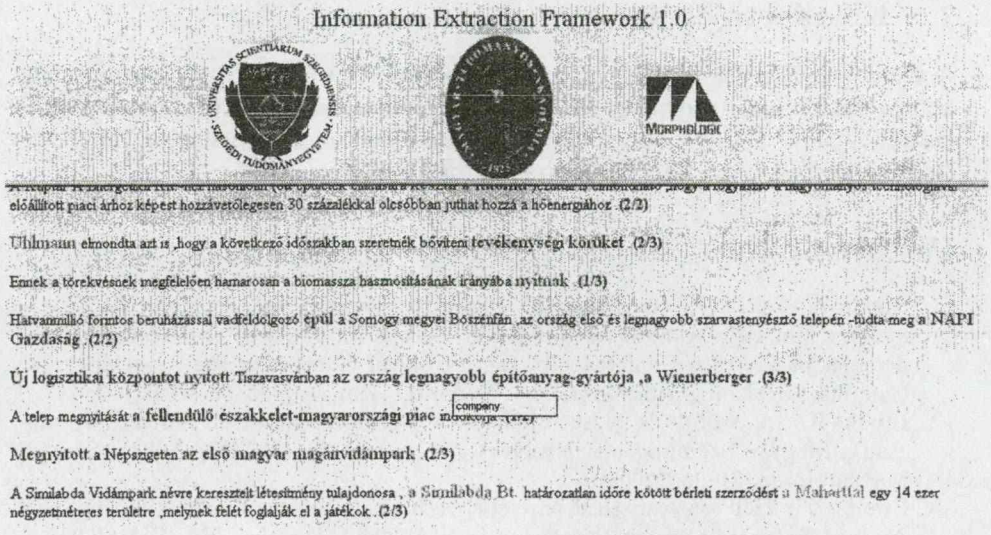
5. A felhasználói felület és a webes megjelenítő modul

A programcsomagban az utolsó modul a felhasználói felület, amely egy HTML nyelvű weblapot készít. Ez egyrészt tartalmazza a beolvasott nyers szöveget, másrészt a programcsomag által hozzáadott metainformációkat. Ez utóbbiakat grafikus eszközökkel jeleníti meg. A modatokban azonosított szereplők különböző színekkel, a szerepek nevei a weblapon lebegő üzenetablakokban (*tooltip*ekben) jelennek meg. A mondatokra illesztett szemantikus keret összes szereplőjének száma és az azonosított szereplők száma a mondatok után található. A 3. ábrán a webes megjelenítő modul egy képernyője látható.

Egy alternatív megjelenítő modul az eredményeket nem weblapon, hanem Excel-ablakban jeleníti meg. Az azonos eseményeket egy munkalapon, a mondatokat egy-egy sorban, az azonos szereplőket pedig azonos oszlopokban jeleníti meg. Ez a táblázat további feldolgozások kiindulópontja lehet.

6. Eredmények

Az NKFP projekt keretében ugyanannak az adatbázisnak és háttértudásnak a felhasználásával a konzorciumon belül két információkinyerésre alkalmas prog-



3. ábra. A webes megjelenítő modul egy képernyője

ramrendszer is készült. Az egyik [12] [13] nem különálló modulokból álló, pragmatikus, célratörő rendszer, amely azonban a folyamatos kísérletezést, a modulok cserélgetését nem támogatja. Az itt bemutatott rendszer e hiányosságokat igyekszik kiküszöbölni, és kifejezetten a további információkinyerési kutatások támogatására készült. Az Informatikai Tanszékcsoport Európai K+F pályázatokkal próbál támogatást szerezni a további kutatómunkához.

A cikkben bemutatott programcsomag tesztelésére a kutatók egy keretrendszert (benchmark) készítettek. Ez kézzel előre annotált, a rendszer számára ismeretlen mondatokat tartalmaz két előre kiválasztott témakörben: a tulajdonosváltás és az új telephely nyitása témakörében. [9] Erre a két témakörre megfelelő számú szemantikuskeret-definíció és rövidhír állt rendelkezésre. A tesztmondatok szöveges alakjára lefuttatták a programcsomag egyes komponenseit, majd összehasonlították a kézzel készített és a gép által előállított két állomány metainformációit. Tekintve, hogy a programcsomag több elemből áll, az egyes modulok hibája kumulálódik a végeredményben. Az eredmények megbízhatóbb értékelése érdekében arra is van lehetőség, hogy az egyes modulok által adott részeredményeket külön értékeljék, és összehasonlítsák az etalonfájllal.

A programcsomag által elért eredmény 70,2% (a pontosságból és a találati arányból számított kombinált érték). [10] A hibák 44%-át a felszíni elemző 29%-át a mondatmintákat felismerő program követte el. Tekintve, hogy a felszíni elemző egyelőre viszonylag sok hibát követ el, ha a hibás főnévi szerkezetek miatt bekövetkező mintallesztési hibákat nem számítjuk, akkor a pontosság jobb: 83,4% lesz.

[9] A benchmarkban 176 rövidhír, illetve 285 mondat szerepel.

[10] $F = \frac{2rp}{(r+p)}$, ahol r a találati arány, p a pontosság.

7. Köszönetnyilvánítás

A szerzők ezúton fejezik ki köszönetüket az OM NKFP 2/017/2001 projektbeli konzorciumi partnereiknek, az MTA Nyelvtudományi Intézet Korpusznyelvészeti Osztályának és a MorphoLogic Kft.-nek, akikkel a tudományos és szakmai kapcsolatokon túl szoros, személyes kapcsolatot alakítottak ki.

Hivatkozások

1. Alexin Z., Csirik, J., Gyimóthy, T., Bibok K., Hatvani, Cs., Prószéky, G., Tihanyi, L.: Manually Annotated Hungarian Corpus. in Proc. of the Research Note Sessions of the 10th Conference of the European Chapter of the Association for Computational Linguistics EACL'03, Budapest, Hungary, 53–56 (2003).
2. Bibok K.: A szóról és a szófajokról (a számítógépes nyelvfeldolgozás kapcsán), Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2003), Szeged, Magyarország, 31–36, (2003).
3. Brants, T.: TnT – a statistical part-of-speech tagger, in Proceedings of the Sixth Applied Natural Language Processing (ANLP-2000), Seattle, USA, WA (2000).
4. Brill, E.: Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. Computational Linguistics, **21** 543–565 (1995).
5. Csendes D., Csirik J., Gyimóthy T.: The Szeged Corpus: A POS Tagged and Syntactically Annotated Hungarian Natural Language Corpus in Proc. of the Seventh International Conference on Text, Speech and Dialogue (TSD 2004), Brno, Czech Republic, 41–47, (2004).
6. Csendes, D., Hatvani, Cs., Alexin, Z., Csirik, J., Gyimóthy, T., Prószéky, G., Váradi, T.: Kézzel annotált magyar nyelvi korpusz: a Szeged Korpusz, Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2003), Szeged, Magyarország, 238–245, (2003).
7. Farkas R., Konczer K., Szarvas Gy.: Szemantikus keretillesztés és az IE rendszer automatikus kiértékelése Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2004), beküldve, Szeged, Magyarország, (2004).
8. Hócza, A.: Noun Phrase Recognition with Tree Patterns elfogadva az Acta Cybernetica c. lapban történő megjelenésre (2004).
9. Kuba A., Hócza A., Csirik J.: POS Tagging of Hungarian with Combined Statistical and Rule-Based methods in Proc. of the Seventh International Conference on Text, Speech and Dialogue (TSD 2004), Brno, Czech Republic, 113–120, (2004).
10. Kuba A., Bakota T., Hócza A., Oravecz Cs.: A magyar nyelv néhány szófaji elemzőjének összevetése Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2003), Szeged, Magyarország, 16–22, (2003).
11. Miháczi András, Németh László, Rácz Miklós: Magyar szövegek temészetes nyelvi előfeldolgozása Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2003), Szeged, Magyarország, 38–43, (2003).
12. Prószéky G.: Automatikus információszerzés gazdasági rövidhírekből. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2003), Szeged, Magyarország, 161–166, (2003).
13. Prószéky G.: Automatikus információszerzés gazdasági-politikai rövidhírekből. VIII. Országos (Centenáriumi) Neumann Kongresszus kiadványa, Budapest, Magyarország, 359–367, (2003).

Szemantikuskeret-illesztés és az IE rendszer automatikus kiértékelése

Farkas Richárd[1], Konczer Kinga[2], Szarvas György[1]

[1] MTA SZTE Mesterséges Intelligencia Tankszéki Kutatócsoport
{rfarkas, szarvas}@inf.-szeged.hu
[2] Szegedi Tudományegyetem
kinga.konczer@hungary.org

Kivonat: Frametagger az SZTE Nyelvtechnológiai Csoportjának szemantikuskeret-illesztő programja, ami a gazdasági rövidhírek szereplőinek azonosítására született. A program az NKFP 2/017/2001 projekt[1] keretében, a Nyelvtudományi Intézet által elkészített, majd az SZTE által bővített keretekre és szemantikus táblázatokra épül. A program a szegedi IEToolChain[2] információkinyerő rendszer végső modulja. Előadásunkban bemutatjuk az IEToolChain kiértékelésére született Benchmark programot is, aminek célja, hogy pontos képet kapjunk arról, hogy az IEToolChain egyes moduljainak javítása, cseréje hogyan befolyásolja az egész rendszer hatékonyságát.

1 Szemantikuskeret-illesztés

Az információkinyerés célja a lényeges információ megjelölése és összegyűjtése dokumentumokból. A működő rendszerek általában megelégszenek a mondatok fontosabb szereplőinek azonosításával (az általános szemantikus szerepcímkézési feladattal [3] szemben, ahol a cél az összes ige vonzatkörnyezetének meghatározása) anélkül, hogy részletes szintaktikai ill. szemantikai elemzést végeznének.

Rendszerünkben a mondat szereplőinek azonításához a mondat ún. felszíni elemzését és egy szemantikuskeret-halmazt használunk fel. A keretek eseményeket írnak le azok szereplőinek szintaktikai és szemantikai megkötésein keresztül. Esetünkben tehát az információkinyerés a keretek célszavának illetve többi szerepének illesztése a mondatra.

1.1 Frametagger

Az Frametagger feladata, hogy az IEToolChain korábbi moduljai által előállított szintaktikailag elemzett szövegeken megtalálja és bejelölje a legjobban illeszkedő szemantikus szerepeket az előre definiált kerethalmaz alapján.

Frametagger inputját tehát a szintaktikailag (mondat- és szószegmentált, szófajilag egyértelműsített, NP taggelt) bejelölt szöveg, szemantikus táblázatok és a kerethalmaz alkotják. A – Nyelvtudományi Intézet által elkészített – szemantikus táblázatok 5471

főnévi és 3972 melléknévi jelenést osztályoznak (osztályok pl.: intézmény, absztrakt, cselekvőképes stb.)

Az általunk használt kerethalmaz a céginformációs gazdasági rövidhírek két témakörét írják le, a tulajdonosváltást és az intézménynyitást. A 71 darab keret szintaktikai és szemantikai megkötésekkel él az egyes szerepekre. A szükséges szemantikai információkat a szemantikus táblázatok alapján tölti ki a program.

Az NKFP 2/017/2001 projekt keretében elkészült kereteket az alábbiakkal bővítettük ki:

1. *Célszó* fogalmának bevezetése. Minden keretben a – korábbi szerepek közül – kijelöltünk pontosan egy célszót. A célszó általában ige (pl.: „megvásárol") de lehet más is, pl.: „alapkő". Egy illesztést csak abban az esetben tekintünk helyesnek ha a célszó illesztésre került, és a célszón felül legalább további egy szerep illeszkedik.
2. A célszavakon kívüli szerepekhez *prioritási* értéket vettünk fel. A szerep prioritási értéke megmutatja, hogy a szerep mennyire fontos az adott keretben a többi szerephez viszonyítva.
3. A szerepekhez különböző *pozícióbeli megkötések*et is adtunk. Azon felül, hogy keretmegszorítások közt megadható, hogy az egyik szerep a másik függvénye (azaz csak akkor illeszthető, ha a másik szerep illesztett), azt is meghatározhatjuk, hogy a függő szerep a függvényhez képest balra, jobbra helyezkedik-e el a mondatban, vagy közvetlen bal ill. jobb szomszédja-e. Erre elsősorban a birtokos illetve egyéb szerkezeteknél van szükség.

Mivel a mondat szavai, szerkezetei és a keretek szerepei egy (n*m-es) hozzárendelési feladatot határoznak meg, célszerű volt, hogy a programot az alábbi egyszerű algoritmus alapján építsük fel:

```
minden(mondatra) {
    minden(keretre) {
        költségmátrix kitöltése;
        magyar módszer végrehajtása;
    }
    legolcsóbb hozzárendelések bejelölése;
}
```

A hozzárendelési feladat kitöltése két részből tevődik össze, először minden (szó;szerep) párra megvizsgáljuk, hogy az adott megkötéseket teljesíti-e, majd a lehetséges illesztésekhez heurisztikaértéket számítunk. A felhasznált *heurisztikák* a következők: prioritási érték, tulajdonnév, mélység a szintaktikai fában.

Az olyan esetek tették szükségessé a mélységheurisztika hozzáadását, amikor a legfelső szintű szintaktikai egység több szerepből áll (pl: „28 százalékos részesedést"). A program szavakat feleltet meg a szerepeknek, de az illesztett szavak helyett azt a legmagasabb szintű nyelvtani szerkezetet jelöli be, amelyiknek az adott szó a feje.

A feladat magyar módszerrel történő megoldása időigényes, viszont az összes lehetséges megoldás által meghatározott térben keres, így nem veszíthetünk el megoldásokat.

1.2 Vizualizáció

A Frametagger outputja egy szemantikai információkkal bővített XML állomány, aminek átlátása a felhasználó számára igen komplikált. Ezért fejlesztettünk egy modult, ami az XML fájlt két felhasználóbarát formátumba konvertálja:

1. Egy HTML fájl generálódik, amelyben a megtalált szerepek különböző színekkel vannak jelölve, a szerep típusa pedig megjegyzésben jelenik meg. Ezen felül miden mondat után táblázatos formában is megjelennek a mondat különböző szerepei.
2. Egy Excel táblázatot is készítünk, amelyben egy munkalapon láthatjuk az azonos témájú híreket. A táblázat sorai egy-egy mondatot, oszlopai az egyes szerepeket tartalmazzák. Ennek segítségével könnyen készíthetünk komplex kimutatásokat (pl.: „Milyen cégeket vásárolt fel az OTP?")

2 Benchmark

Miután összeállt az egységes szegedi IEToolChain információkinyerő modullánc tudatában voltunk, hogy az egyes modulok külön-külön (tökéletes bejövő adatok mellett) milyen helyesen működnek, de nem tudtuk, hogyan befolyásolják a rendszert, mint egységet vizsgálva.

Egy olyan eszközt fejlesztettünk ennek vizsgálatára, ami egy etalonhoz hasonlítva nemcsak a végeredményről közöl (pontossági és találati) értékeket, hanem megpróbálja a helytelen (nem a legmegfelelőbb) illesztéseknél meghatározni, hogy mi a hiba oka és így melyik modul okolható érte.

Etalonnak a Szeged Korpusz NewsML részkorpuszából [4] 176 db hírt (285 mondatot) leválasztottunk. A – szintaktikailag már korábban annotált – mondatokat a kerethalmazhoz igazodva szemantikailag is bejelöltük. Ezt a mondathalmazt kivettük az összes tanuló algoritmust használó IEToolChain modul tréninghalmazából, így az tekinthető ismeretlen szövegnek.

A kiértékeléshez az alábbi hibakategóriákat határoztuk meg:

1. **Topikhiba**: ha az illesztett keret nem abba a témakörbe tartozik, mint a bejelölt keret.
2. **Feleslegesen felismert szerep**: olyan szerepek, amelyeket a gépi elemzés bejelölt, viszont az etalonbeli mondatban nem szerepelnek.
3. **Mondatszegmentálási hiba**: a program azért illesztette a szerepet helytelenül, mert az etalonbeli szereplőt külön mondatba szeparálta a mondatszegmentáló modul.
4. **POS hiba**: azért nem sikerült az illesztés, mert a helyes szerep MSD kódja nem egyezik meg a releváns helyeken a gépi elemző által adott kóddal.
5. **Lefedés**: azért sikertelen az illesztés, mert egy másik szerep eltakarja a felismerendő szavakat. Ez tulajdonképpen a fedő szerep hibája.
6. **NP hiba**: akkor tekintünk egy hibát NP hibának, ha a bejelölt illetve felismert szerepek közül az egyik a másik részhalmaza.

7. **Tagmondathiba**: a felismert szerep másik tagmondatba esik, mint a célszó. (az etalonban jelezve vannak a tagmondathatárok, viszont IEToolChainben nincs tagmondat-határolás)
8. **Igekötőhiba**: a gépi elemzés ugyan megtalálta az igét, de annak elváló igekötőjét nem jelölte be.
9. **Egyéb hiba**

A program az etalonbeli mondatokhoz hasonlítja egy TEI[5] kódolásnak megfelelő fájlhalmaz mondatait. Így a program megteremti a platformot arra is, hogy különböző magyar (gazdasági híreket feldolgozó) információkinyerő rendszereket, illetve azok moduljait (részfeladatokat végrehajtó egységeit) összehasonlíthassuk.

3 Eredmények és jövőbeni tervek

Az előző fejezetben bemutatott módszertan alapján a szegedi IEToolChain rendszer 70,2% pontossággal és 70,3% találati aránnyal működik. A két legjelentősebb hiba (és hibákon belüli arányuk) az NP hiba; 44% és a felesleges szerep; 29%. Mindössze 1 mondatnál követ el topikhibát a gépi elemzés (két témakör esetén).

Ha az illesztés jóságát másképp definiáljuk, és részleges egyezéseket (NP hibás illesztések tulajdonképpen a helyes szerepre találnak rá, csak nem ismerik fel azt pontosan) is elfogadjuk jó illesztésnek, akkor IEToolChain 83,4% F mértéket[1] produkál. Ezek alapján jogosan jelenthetjük ki, hogy a szegedi információkinyerő rendszer jelentős időt takaríthat meg – mint előfeldolgozó – egy manuális elemző számára.

Jelenleg folyamatban van a keretbeli -keretekben már szereplő- pozíciómegkötések Frametaggerbe történő beépítése, valamint az egyes részfeladatok alternatíváinak modulláncbeli tesztelése. Ezektől a javításoktól az IEToolChain további javulását várjuk.

A jövőben szeretnénk a Frametagger elé egy témaosztályozó modult beilleszteni. Ugyanis – mint az a 1.1 fejezetben látható – jelenleg a kerethalmazban nincs semmilyen különbség a két – jelenleg keretekkel lefedett – témabeli keretek közt. Azaz tulajdonképpen a megtalált keret azonosítja a témakört. A témák (elő)osztályozására feltétlenül szükség lesz, amikor a témakörök száma emelkedni fog.

Most végezzük ezen felül a teljes szintaxis felismerését végző modul integrálását az IEToolChainbe. Ez felveti a kérdést, hogy a kézzel kialakított Benchmark-hibaosztályok meddig és milyen áron bővíthetőek. Az elkövetkezendőkben szeretnénk megvizsgálni, hogy az általános, fatávolság alapú összehasonlítások versenyezhetnek-e a Benchmark specialitásokat kihasználó összehasonlításával.

[1] Az F mérték a pontossági és találati arány harmonikus közepe.

Bibliográfia

1. Prószéky Gábor: Automatikus információszerzés gazdasági rövidhírekből. MSzNy 2003 (2003) 161–166
2. Alexin Zoltán, Gyimóthy Tibor, Csirik János: Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2004), beküldve, Szeged, Magyarország, (2004).
3. Xavier Carreras and Lluís Márques: Introduction to the CoNLL-2004 Shared Task: Semantic Role Labeling. Proceedings of CoNLL-2004 (2004) 89–97
4. Csendes Dóra, Csirik János, and Gyimóthy Tibor: The Szeged Corpus: A POS Tagged and Syntactically Annotated Hungarian Natural Language Corpus. In Sojka et al. [SKP04], pages 41–47.
5. Oravecz, Cs., Váradi, T.: TEI Encoding of the Hungarian Explanatory Manual Dictionary. In Kiefer et al. (eds.) Papers in Computational Lexicography COMPLEX'99, 1999, pp. 229-236

Információkinyerés igeneves szerkezetekből

Gábor Kata[1], Héja Enikő[1], Mészáros Ágnes[1]

MTA Nyelvtudományi Intézet,
H-1399 Budapest VI. Benczúr u. 33. Pf. 701/518
e-mail: {gkata,eheja,magnes}@nytud.hu

Kivonat Előadásunkban a NewsPro információkinyerő rendszer egy továbbfejlesztési lehetőségét mutatjuk be. A NewsPro egyik hiányossága, hogy csak igei állítmánnyal kifejezett eseményeket ismer fel, az igenévvel kifejezett eseményekre nem tudja illeszteni a szemantikai kereteket. Így a felhasználó az információ egy részéhez nem fér hozzá, valamint – mivel mondatonként csak egy eseményt ismer fel a rendszer – az események közti összefüggések is gyakran rejtve maradnak. Ennek kiküszöbölésére egy előfeldolgozó modult fejlesztettünk ki, mely az igeneves szerkezeteket teljes propozícióvá alakítja, így a szemantikai keretek minden további átalakítás nélkül illeszthetők ezekre.

1. Bevezetés

Az alábbiakban egy olyan nyelvészeti témájú alkalmazott kutatást szeretnénk bemutatni, melynek célja, hogy a szabályalapú információkinyerés hatékonyságát növelje. Munkánk az NKFP 2/017/2001 projektumban a MorphoLogic Kft., a Szegedi Egyetem Informatikai Tanszékcsoportja és az MTA Nyelvtudományi Intézet Korpusznyelvészeti Osztálya által elkészített NewsPro információkinyerő rendszer [1] továbbfejlesztését célozza. A NewsPro rendszer a bemeneti szövegen részleges szintaktikai elemzést hajt végre, majd előre definiált szemantikai kereteket, azaz eseménymintákat illeszt a szövegre. Sikeres illesztés esetén az eseménymináták a szöveg elemeivel feltöltődnek, így a kimenet azonosítja a hírben szereplő eseményt, valamint annak szereplőit, attribútumait és körülményeit. A rendszer fejlesztésekor a vállalati rövidhírekre összpontosítottunk, így az eseménysablonok ezt a területet fedik le, de természetesen a program alkalmassá tehető tetszőleges tematikájú hírek kezelésére. A vállalati rövidhírek az MTI archívumából származnak. Egy hír általában egy mondatból áll. A hírekre illesztett eseménymináták központjában ragozott igék állnak, melyek bővítményei képviselik az ige által kifejezett esemény szereplőit, körülményeit, attribútumait. A mintaillesztés tehát a szintaktikai elemző által állítmányként megjelölt igéből, illetve annak vonzatkeretéből indul ki. Emögött az az implicit feltételezés áll, hogy a hírben egy igei állítmány fejezi ki a fő eseményt. Ez a megközelítés, bár a hírek nagy részénél hatékonyan működik, gyakran azzal az eredménnyel jár, hogy a *másodlagosnak*, ismertnek feltételezett információkat, melyek többnyire

a fő esemény előzményeként, okaként vannak feltüntetve, kihagyja a mintaillesztésből. Ezek a másodlagos információk ugyanis nem ragozott igék, hanem igéből képzett főnevek vagy igenevek formájában szerepelnek a szövegben. Például:

(1)

A gyártók által tegnap bejelentett árcsökkentések és a hitelkamatok mérséklése nyomán megnőtt a kereslet az új autók iránt.

Noha a fenti mondat központi információja a kereslet növekedése, hírértékkel bírhat az árcsökkenés is. Előfordulhat, hogy a felhasználó nem olvasta a korábbi híreket, vagy kíváncsi az események közti összefüggésekre, melyek akkor tárhatók fel, ha a rendszer képes egy hírben több eseményt is elemezni. A megoldandó feladat fontosságát jelzi, hogy az MTI rövidhírekből álló 25,902 mondatos korpusz összesen 6,567 folyamatos vagy befejezett melléknévi igeneves szerkezetet tartalmaz.

A jelenség kezelését a NewsPro rendszerben egy előfeldolgozó modul feladataként képzeltük el. A modul az igeneves szerkezeteket ragozott igét tartalmazó mondattá alakítja. Az így átalakított szövegen a nyelvtani elemzés és a szemantiaki keretek illesztése külön változtatás nélkül alkalmazható. Első lépésben csak a főnévi csoportokon belül előforduló befejezett melléknévi igenevekkel foglalkoztunk. Feltételeztük, hogy az igéből képzett melléknévi igenevek átalakíthatók ragozott igét tartalmazó propozícióvá, mert az igenév megőrzi az alapige jelentését, és argumentumai (legalábbis azok egy része) levezethetők a főnévi csoport szerkezetéből. A befejezett igenév mindig előidejű, így az ige múlt idejű lesz. Az átalakított mondatokon a mintaillesztés várhatóan még nagyobb pontossággal működik, mint az érintetlenül hagyott szövegrészeken, mivel a transzformáció során lehetőségünk van meghatározni a kimeneti mondatban a mondatrészek sorrendjét[1]. Ez pedig megkönnyíti a szintaktikai elemzést és az erre épülő eseménysablon-illesztést.

Az előfeldolgozó transzformáció sikere természetesen nem csak azon múlik, hogy hogyan sikerül az igeneves szerkezet szintaxisából levezetni a propozíciót, hanem azon is, hogy mekkora információtartalma van az így képzett mondatoknak. Kísérletet tettünk arra, hogy kialakítsunk egy olyan algoritmust, mely kizárólag szintaktikai információ alapján kiszűri a vélhetően informatív szerkezeteket.

A következő bekezdésekben először bemutatjuk azt a korpuszfeldolgozó eszközt, melyet a szabályok megírásához és teszteléséhez használtunk (2.). Ezt követően leírjuk az informatív szerkezetek kiszűrésére használt algoritmust (3.), majd részletesen ismertetjük a szabályokat (4.), végül kitérünk a szabályok tesztelésének eredményére (5.).

[1] A kimeneti mondatok elemeinek toldalékolásával egyelőre nem foglalkoztunk, ám - mivel kevés morfológiai változtatásra van szükség - ezt viszonylag rövid időn belül megoldhatónak gondoljuk.

2. A korpuszfeldolgozó eszköz

A transzformációt végző szabályok elkészítéséhez és teszteléséhez, valamint a szöveg szükséges előfeldolgozásával kapcsolatos valamennyi feladathoz az Intex nevű, kutatási célokra szabadon használható korpuszannotáló szoftvert [2] használtuk. Az Intex alapvetően lexikalista megközelítésű nyelvelemzésre alkalmas, alappillére az erre a célra kialakított szótár, mely egy szinten kódolja a morfoszintaktikai és a szemantikai információt, így az a nyelvtani elemzés minden szintje számára hozzáférhető. Ez nagy előnyt jelentett számunkra a transzformációt végző nyelvtan írásakor, hiszen – amint a következő fejezetekből kiderül – nyelvtanunknak hivatkoznia kellett az igenevek alapigéjére (amit szintén a szótárban kódoltunk), valamint az alapige szintaktikai jegyeire is.

3. Melyek az informatív igenevek?

Az adatok vizsgálata során kérdésként merült fel, hogy mikor érdemes egy befejezett melléknévi igenevet igévé alakítani. Egyfelől nem elhanyagolható a főnévi csoport által hordozott információtartalom, amely annál nagyobb, minél több bővítménye van jelen az igenévnek. Már ez elégséges indok arra nézve, hogy csak a bővítménnyel rendelkező igenevekkel foglalkozzunk. Azonban a fentieken túl sokkal komolyabb problémák is felmerülnek a bővítménnyel nem rendelkező igenevek igévé alakítása kapcsán. Ezt illusztrálják az alábbi NP-k és a szabályaink kimeneteként kapott mondatok:

(2)

a **jegyzett** tőke	[particip Valaki jegyzett tőke -t]
a **nyomott** hangulatot	[particip Valaki nyomott hangulatot -t]
a **mérsékelt** PC-chip kereslet	[particip Valaki mérsékelt PC-chip kereslet-t]
a **nyomtatott** sajtóban	[particip Valaki nyomtatott sajtóban -t]
a **ragozott** szóalakokból	[particip Valaki ragozott szóalakokból -t]
a **kerekített** euróárak	[particip Valaki kerekített euróárak -t]
a **használt** ingatlanok	[particip Valaki használt ingatlanok -t]

A fenti igenevek esetén a rövidesen ismertetésre kerülő átalakítási szabályok nem jól működnek. Ennek okát abban látjuk, hogy a szóbanforgó kifejezések valójában nem igenevek, hanem melléknevek, és a szófajváltással együtt a vonzatstruktúrájuk és a jelentésük is megváltozott. Így a kapott propozíciók helytelenségére két – egymástól nem független – magyarázatot adhatunk. Ha a jelentésváltozás egyértelmű (pl.:'*nyomott hangulat*'), a kiinduló ige jelentése nem releváns az NP jelentése szempontjából, így az eredeti igével való behelyettesítés szemantikailag helytelen mondatokat eredményez. Azon melléknevek esetében, ahol a jelentésváltás kevésbé éles, az igévé való visszaalakítás után azért kapunk szemantikailag helytelen mondatokat, mert – feltételezésünk szerint – a

melléknévvé válás során az eredeti ige teljes vonzatstruktúrája törlődik . Így tehát az eredeti ige alanyi argumentumhelyén megjelenő főnévnek nincs szemantikai szerepe a melléknevet tartalmazó NP-ben. Azaz *'a nyomtatott sajtó'* esetén nem az a fontos, hogy valaki kinyomtatta azt a sajtóterméket, hanem az, hogy ez most már ilyen állapotban található. Hasonló a helyzet a *'kerekített euróárak'*-kal, a *'ragozott szóalakok'*-kal és a *'használt ingatlanok'*-kal is.

Az általunk kifejlesztett szabályrendszer alapja az a hipotézis, hogy csak a bővítménnyel rendelkező *'ige+(t)t'* alakú kifejezéseket tekintjük igeneveknek és a hasonló képzővel ellátott, ám bővítmények nélküli igéket melléknéveknek. Bővítmények alatt a kötelező vonzatokat vagy a szabad határozókat értjük. Így a (2)-ben szereplő NP-k kívül esnek vizsgálódásunk körén. Az alábbiakban felsorolunk néhány kritériumot, amelyek lehetővé teszik a melléknevek és az igenevek elkülönítését[3]:

(a) Predikatív helyzetben, fokozott formában csak melléknév fordulhat elő. Ezen teszt alapján levonhatjuk azt a következtetést, hogy példamondatainkban melléknevek szerepelnek[2]. Sok esetben ugyan – szemantikai okok miatt – nem fokozhatóak (pl.: **nyomtatottabb*), de minden esetben kerülhetnek állítmányi pozícióba[3].
(b) Továbbképzéssel csak melléknévből képezhető határozószó. A lexikalizálódott alakoktól eltekintve az összes (2)-ben szereplő kifejezésből képezhető határozószó[4]. Így ez a kritérium is azt támasztja alá, hogy a szóbanforgó esetekben melléknevekről van szó.
(c) Csak az igenevek előtt van elváló igekötő, a melléknevekben található igekötők nem válhatnak el[5].

Bár ez utóbbi szempont vajmi keveset árul el az eddig tárgyalt kifejezések szófajáról, mivel egyikőjük alapigéje sem rendelkezik igekötővel, ez a kritérium nemsokára még nagy segítségünkre lesz. Azt állítjuk, hogy ha egy megfelelő formájú ige környezetében azt módosító szabad határozót találunk, ez már elégséges alapot nyújt arra nézve, hogy az adott kifejezést igenévnek tekintsük[6], azaz nem szükséges vonzat megőrzése az igenéviséghez. Ezt az elgondolásunkat (a), (b), (c) disztribúciós feltételek alátámasztják:

(a') *„*A múlt héten mérsékeltebb PC-chip kereslet'* és *„*A PC-chip kereslet [a múlt héten mérsékelt] volt'* [7].

[2] Bár bizonyosos esetekben lexikalizálódott kifejezésekkel van dolgunk, amelyek a kritériumoknak nem megfelelően viselkednek (pl.: **'jegyzettebb tőke'*)

[3] Pl.: *'A hangulat nyomott volt.'*, *'A PC-chip kereslet mérsékelt'*, *'A magyar sajtó zöme nyomtatott'* (és nem elektronikus).

[4] *'A mérsékelten csökkenő PC-chip kereslet'* vs. **'Az EU által mérsékelten csökkenő PC-chip kereslet'*; *'A használtan vásárolt ingatlanok'* vs. **'Az árusításra használtan vásárolt ingatlanok'*.

[5] **'A budai áruházak fel nem újítottak'* vs. *'az állam által fel nem újított utak'*.

[6] Itt Komlósy(1992) nézetével vitatkozunk, aki szerint az igenéviséghez szükséges az alapige vonzatainak megőrzése.

[7] Szerkezeti homonímia elkerülése érdekében ahol szükséges szögletes zárójellel jelöltük az összetevőket. Ha *'a múlt héten'* a *'mérsékel'* modósítója, akkor (a') (b')

(b') *„*A [múlt héten mérsékelt]en csökkenő PC-chip kereslet*" vs „*A múlt héten mérsékelten csökkenő PC-chip kereslet*".

Tehát (a) és (b) alapján beláttuk, hogy jogos befejezett melléknévi igenévnek tekinteni minden '*ige+(t)t*' formájú kifejezést, ha bármilyen bővítményét (kötelező vonzat, szabad határozó) azonosítani tudjuk. Továbbá – ha közvetve is – de (c) is ezt támasztja alá; ha belátnánk, hogy az igekötők szabad határozók, akkor ennek egyik – szükséges – alapja az a megfigyelés lenne, hogy az esetek többségében az igekötő az igétől viszonylag függetlenül mozog. Mivel az igekötő az igenevek esetében válik el, ekkor viselkedik szabad határozóként. Mivel megfigyeléseink szerint a szabályok igekötős, vagy egyéb bővítménnyel rendelkező igenevek esetében működtek jól, ez fenti állításunk közvetett bizonyítékát jelenti. Így ebben a részben már csak egy feladatunk maradt, indokokkal szolgálni arra nézve, hogy miért tekintjük az igekötőket szabad határozóknak. Első pillantásra furcsának tűnhet, hogy miért jogos az igekötőket az ige bővítményei közé sorolni, hiszen az igekötő és az alapige egy lexikai tételt,[8] és ha az igekötő közvetlenül az ige előtt van, akkor egy fonológiai szót is alkotnak. Azonban a *lexikai integritás* elve alapján nincsen olyan szintaktikai szabály, amelynek bemenetéül egy szó részei szolgálnának[4]. Ezzel szemben az igekötők egy mondaton belül viszonylag függetlenül mozoghatnak az igétől, tehát vannak olyan szintaktikai szabályok, amelyeknek a bemenetét igekötők képezik. Ebből következik, hogy az igekötős ige nem lehet összetett szó. Továbbá, az igekötőkhöz disztribúciós szempontból hasonlóan viselkedik a bővítményeknek egy szintaktikailag nem egységes osztálya[9]. Ez arra utal, hogy az igekötőnek vagy vonzatnak kell lennie, vagy szabad határozónak. Most már csak az a kérdés, hogy melyiknek tekintsük őket. Komlósy(1992) szerint ha az igekötő az igének vonzata, akkor – igaz ugyan, hogy egy függvény-szerű kifejezésből függvény-szerű kifejezéseket kapunk – az igekötő maga nem lehet függvény, de azok a kifejezések, amelyek nem függvények, mindig (individuumra, tényállásra) referáló kifejezések kell, hogy legyenek, valamint formailag mindig maximális főkategóriákkal vannak kifejezve.
Az igekötőkre ezek egyike sem áll. Ezek tehát azok a megfontolások, amelyek alapján úgy döntöttünk, hogy

1. Igenévnek tekintünk minden nemcsak vonzattal rendelkező megfelelő formájú kifejezést, hanem azokat is, amelyek környezetében csak szabad határozó van jelen.
2. Szabad határozónak tekintjük az igekötőket is, így a csak igekötővel rendelkező formák is a szabályok bemenetét képezik.

Így alátámasztottnak tekintjük kiinduló hipotézisünket, mely szerint csak igekötővel vagy egyéb bővítményekkel rendelkező kifejezéseket tekintünk igeneve-

kritériumok valóban azt mutatják, hogy szabad határozóval módosított ige+t formájú kifejezés igenév.

[8] Az igekötő-ige egység együtt képezi szóképzés bemenetét.

[9] '**Pirosra** *festi a kerítést*'; '*Péter* **ügyesen** *vezeti a labdát*'; '*Péter* **okosnak** *tartja Marit*'; '*Péter* **úszni** *akar*'. Ezeknek az eltérő szófajú szavaknak egy része az ige vonzata, egy másik része pedig szabad határozója.

knek és azokat, amelyek környezetében ezek egyike sem fordul elő, melléknéveknek. Mivel eredeti célunk az volt, hogy kiszűrjük az informatív szerkezeteket, azt kell megvizsgálnunk, hogy a szintaktikai krtitériumok által elkülönített két csoport hogyan állítható párhizamba az informatív – nem informatív csoporttal. Azt látjuk, hogy az általunk informatívnak tartott szerkezetek egybeesnek a fenti szintaktikai kritériumokkal definiált igeneves szerkezetekkel.
A következő pontban a szabályokat fogjuk részletesen bemutatni.

4. A nyelvtan

Az NP-n belüli melléknévi igeneves szerkezetek transzformációs szabályainak kialakításakor az alábbi alapfeltételezésekkel éltünk:

(a) melléknévi igenevet tárgyas és tárgyatlan igéből is lehet képezni,
(b) tárgyatlan ige esetén az NP fejét alkotó főnév a melléknévi igenév alapigéjének alanya,
(c) tárgyas ige esetén az NP fejét alkotó főnév a melléknévi igenév alapigéjének tárgya; ebben az esetben az alapige ágens alanyú,
(d) a melléknévi igenév előtt megjelenhetnek az alapige vonzatai és szabad határozói (ragos NP-k, főnévi igenév, melléknévi csoport, határozószók stb.),

valamint – bár nem feltételezhetjük, hogy minden, igenevet tartalmazó NP elején áll determináns – a kezelni kívánt főnévi csoportok körét leszűkítettük a determinánssal kezdődő NP-kre. Erre azért volt szükség, mert a melléknévi igenév előtt megjelenő, igétől örökölt vonzatok igen sokfélék lehetnek, így determináns nélkül rendkívül nehéz lenne az igenevet tartalmazó főnévi csoport bal szélét pontosan meghatározni (a szerkezeti homonímia gyakorisága miatt ez világismeret nélkül gyakran lehetetlen). Így azonban feltételezhetjük, hogy minden, a determináns és az igenév között megjelenő elem az igenév alapigéjének bővítménye, míg az NP fejét képviselő főnév saját bővítményei az igenév mögött találhatók. Például az " a*kulcsfontosságúnak* tekintett *német* eladásoknak" főnévi csoportban a *kulcsfontosságúnak* az igenév, a *német* a főnévi fej módosítója.

A fenti általánosítások alapján tehát először két csoportot különítettünk el: a tárgyas és a nem tárgyas igékből képzett igenevet tartalmazó NP-ket. A transzformációt végző lokális nyelvtanok olyan szótárra támaszkodnak, melyben kódolva van az ige tárgyas ill. tárgyatlan volta[10](tárgyasnak tekintettünk minden olyan igét, melynek lehet tárgyas előfordulása).

Tárgyas igék

A tárgyas alapigéből képzett igenevek átalakításához használt szabály alapja az alábbi transzformáció:

`Det (V_bőv) VMIB N → Valaki V_vmib Det N -t (V_bőv).`

[10] A szótár kialakításához, azaz a szintaktikai viselkedést kódoló jegyekhez a Korpusznyelvészeti Osztályon készült igei vonzatkeret-adatbázist használtuk.

Ahol *Det*: az NP determinánsa, *V_bőv*: az alapige bővítményei, *VMIB*: az igenév, *N*: az NP feje, *V_vmib*: az alapige, a zárójel pedig opcionalitást jelent. Ilyen átalakításra példa:

(3) *a garéi hulladéklerakó ügyében benyújtott keresetét*
particip Valaki benyújtott a kereset -t a garéi hulladéklerakó ügyében. particip

Az alapige argumentumszerkezetét tehát úgy töltjük fel, hogy a főnévi csoport fejét tekintjük tárgynak, az alanyt pedig – ami az esetek többségében nem jelenik meg a szerkezetben – „valaki" névmással töltjük ki, mivel tudjuk, hogy ágens szerepű. Természetesen van olyan eset, amikor az alany megjelenik az „által" névutóval az igei bővítmények szokásos helyén. Az ilyen szerkezeteket az alábbi szabállyal alakítjuk át:

`Det Nsubj által (V_bőv) VMIB N → Nsubj V_vmib N -t (V_bőv).`

Például:

(4) *a bankok által felszámított túl magas hitelkamatok*
[particip bankok felszámított túl magas hitelkamatok -t . particip]

Az alapige alanya nemcsak az „által" névutós szerkezetben jelenhet meg az igenév előtt, hanem alanyesetben is, méghozzá az igenevet tartalamzó főnévi csoport fejének birtokosaként. A birtokos megjelenése önmagában nem cáfolja feltevésünket, mely szerint az igenév előtt megjelenő elemek az alapige bővítményei, hiszen a birtokost többnyire jogosan emeljük alanyi pozícióba:

(5) *a svéd Networks tervezett adósságátalakítási programjában*
[particip svéd Networks tervezett a adósságátalakítási programja -t. particip][11]

Tárgyatlan igék

Tárgyatlannak azokat az igéket tekintettük, melyeknek az igei vonzatkeret-adatbázisban egyetlen tárgyas argumentumszerkezete sem szerepel. A tárgyatlan alapigék argumentumszerkezetének meghatározása nem jelent problémát: az NP feje a tárgyatlan ige alanyával azonos, a többi bővítmény pedig – a tárgyas igéknél látottakhoz hasonlóan – az igenév előtt áll. Érdekes, hogy a rövidhír-korpuszban szereplő, tárgyatlan alapigéből képzett igenevek alapigéje mindig

[11] Sajnos akadnak olyan esetek is, amikor csak a világismeretünk segítségével dönthetjük el, hogy az NP fejének birtokosa azonos-e az alapige alanyával:

a cseh Komercni Banka meghirdetett 60 százalékára
[particip cseh komercni banka meghirdetett 60 százaléka-t particip]

páciens alanyú[12]. Nagyrészt keletkezést, illetve állapotváltozást jelentő igéket találunk köztük. A tárgyatlan igéből képzett igeneveket az alábbi szabállyal alakítjuk át:

`Det (V_bőv) VMIB N → DET N V_vmib (V_bőv).`

Például:

(6) *A kereskedés utolsó perceiben bekövetkezett áremelkedés*
participip A áremelkedés bekövetkezett kereskedés utolsó perceiben. participip

Mint a fenti példából is látható, a tárgyatlan igék argumentumszerkezete maradéktalanul kitölthető az igeneves szerkezet elemeivel. Az információkinyerés szempontjából azonban ezek a transzformációk kevésbé hasznosak, kevesebb implicit információt fejtenek ki, mivel az igenevek olyan igékből származnak, melyek szemantikailag kevéssé tartalmasak: *'bekövetkezett', 'beindult', 'létrejött', 'kialakult', 'megszületett'* – így valószínűleg argumentumaik azonosítása sem nyújt többletinformációt. Ennek ellenére érdemes lehet foglalkozni velük, mivel legalább a már ismert események közti összefüggések feltárásában segíthetnek.

5. Értékelés

A szabályok helyes működésének ellenőrzésére kétféle lehetőség kínálkozik. Egyrészt vizsgálhatjuk az igeneves szerkezetek felismerésének arányát (recall) és a kimenet helyességét (precision). Ezt a folyamatot sajnos részben sem tudtuk automatizálni, mert a tesztkorpusz rendelkezésünkre álló kézzel annotált változatában a melléknévi igenevek nincsenek megkülönböztetve a melléknevektől. Másrészt tesztelhetjük azt is, hogy a modul használata mennyivel növeli a sikeresen illesztett szemantikai minták számát. Az értékelés első lépéseként kézzel ellenőríztük a tesztszövegen kapott találatok egy részét. Ebben a részben a típushibákat mutatjuk be.

Az ellenőrzéshez összesen 7058 mondatot (a teljes korpusz 43%-át) vizsgáltunk meg. A tesztkorpuszban a rövidhírek téma szerint sorrendezve szerepelnek, ezért az ellenőrzött korpuszt úgy állítottuk össze, hogy a teljes korpuszból véletlenszerűen 15, egyenként körülbelül 500 mondatból álló részletet vágtunk ki.

Az alábbi típushibákkal találkoztunk:

1. Helytelen morfológiai elemzés, azaz szótárhiba okozta a hiányok túlnyomó többségét.
2. A nem determinánssal kezdődő NP-ket - amint azt a Bevezetésben is említettük - nem tudjuk kezelni. Szerencsére azonban az informatív (és egyben hosszabb) szerkezetek többsége tartalmaz determinánst.
3. A számneves kifejezéseket (mint például a dátum, pénzes kifejezések, mennyiségjelölők)a szabályok jelen állapotában nem kezeljük tökéletesen. E hiány korrigálására a későbbiekben teszünk kísérletet.

[12] Ez nem jelenti azt, hogy más szövegben sincsenek ágens alanyú tárgyatlan igéből képzett igenevek, pl. *'a társaság lemondott elnöke'*.

4. A szöveg jellegéből fakadóan sok találatban szerepelnek szokatlan NP-k (márkanevet, illetve cégnevet tartalmazó, N N szerkezetű NP-k), melyek felismerése néha problémát okoz.
5. Egyes lexikalizálódott igenevek, bát tartalmazhatnak igekötőt vagy egyéb bővítményt, inkább melléknévként értelmezendők (pl.: *'elmúlt, ismert'*).

Az általunk készített modul a NewsPro rendszer hatékonyságát hivatott növelni, így ennek fényében érdemes a működését értékelni. A fent felsorolt hibák elsősorban a találati arányt rontják, viszont a találati pontosság a nyelvészeti megalapozottság miatt kielégítő. Ez utóbbit fontosabbnak tartjuk, mivel az információkinyerésben a helyes kimenet létrehozása az elsődleges, hiszen a pontatlan találat félrevezetőbb a felhasználó számára, mint a találat hiánya.

Hivatkozások

1. Prószéky G.: Automatikus információszerzés gazdasági rövidhírekből. In: Alexin Zoltán - Csendes Dóra (szerk.): A Magyar Számítógépes Nyelvészeti Konferencia 2003 rendezvényen elhangzott előadások kötete, Szegedi Tudományegyetem Nyomdája, 2003. Szeged, 161-167.o.
2. Silberztein,M.: Dictionnaires électroniques et analyse automatique de textes: Le systeme Intex. Masson, 1993. Paris
3. Komlósy A.: Régensek és vonzatok. In: Strukturális magyar nyelvtan I. Akadémiai Kiadó, 1992. Budapest, 299-529.o.
4. É. Kiss K.: Mondattan. In: Új magyar nyelvtan. Osiris, 1999. Budapest 17-184.o.

A számítógépes terminológiakivonatolás új megközelítése

Kis Ádám
SZAK Kiadó Kft.
adam.kis@szak.hu

Kis Balázs
MorphoLogic Kft.
kis@morphologic.hu

Pohl Gábor
Pázmány Péter Katolikus Egyetem,
Információs Technológiai Kar
pohl@itk.ppke.hu

Az ideális terminológiakivonatoló rendszer emberi beavatkozás nélkül, ismeretlen forrásszövegből, nagy fedéssel és pontossággal kiemeli a terminus technicusnak tekinthető szavakat és kifejezéseket. A terminológia előfordulásának szemantikai, szintaktikai és szövegtani jellemzői egyaránt vannak, így ez a feladat is a számítógépes nyelvészet jellemző modellezési problémájával találkozik: a legtöbb nyelvi jelenséget felszíni jegyeknek kell megfeleltetni.

Ez az előadás felvet egyes, a terminológiakivonatolással kapcsolatos definíciós problémákat, s ismertet egy, új megközelítést alkalmazó terminológiakivonatoló eszköz kifejlesztésére irányuló projektet. Az itt alkalmazott modell a terminológia felismerését a terminus technicusok két alapvető jellemzőjére: a terminológiai helyzetre és a terminológiai szerepre vezeti vissza. Kitér a terminus technicusok hálójának ábrázolására, hangsúlyozza a terminológiai tér nyelv- és témafüggő jellegét.

Az előadás az itt leírt modell irodalmi példákkal való összevetésével és a lehetséges értékelési eljárások felvázolásával zárul.

1. Definíciós kérdések

A terminológiát általában a vele szemben támasztott követelmények határozzák meg. Ezek általában a szabványosítási folyamatokból, illetve a szakfordítás köréből származnak. Megfigyelésünk szerint azonban a terminológiát tartalmazó szövegek – pontosabban azokon belül a terminológiahasználat – nem mindig felel meg ezeknek a feltételeknek (monoszémia, egyalakúság, rendszerszerűség stb.). A legtöbb esetben ugyanis – különösen az új szakterületeken – a terminológia kialakítása intuitív folyamat, s nagyon sokszor metaforikus. A szakterület intézményesülését kísérő szabványosítás pedig sokszor elfogadja, integrálja a „hagyomány" szerint kialakuló kifejezéseket. Ha mégsem, fennáll a veszélye, hogy a preskriptív terminológia nem terjed el (így például a számítógép-hálózati technológiák sem használják mindenben az ISO-szabvány kifejezéseit). A terminológia tehát nem szabványosítási, hanem bonyolult pszicho- és szociolingvisztikai folyamatok eredménye.

A számítógépes nyelvészet – pontosabban alkalmazása, a nyelvtechnológia – nem tud mit kezdeni a preskriptív nyelvleírásokkal, ha az a célja, hogy széles körben használható alkalmazásokat hozzon létre, amelyek valamilyen valós problémát oldanak

meg. A következőkben ezért megkíséreljük a terminológia deskriptív definiálását, amelyből kiindulhat a modellalkotás.

1.1. Terminológiai magatartás

A terminológiai magatartás a szakszövegírók törekvése a szaknyelvi követelményeknek megfelelő fogalmazásra. Meglehet, ez a meghatározás nem igazán teljes, és nem is nagyon fontos, azonban alkalmas illusztrációnak tűnt a mondanivalónk bevezetéséhez. A szakszövegíró így viselkedik: exponál egy fogalmat és meghatározza. Természetesen az ilyen helyzetben született definíciók alkalmiak, érvényük ritkán terjed túl az adott mű határán, azonban módot adnak arra, hogy a szerző az exponált lexémát a továbbiakban terminus technicusként használja, terminológiai szöveget hozzon létre.

A szakszöveget használva a kibocsátó a specifikus fogalmakat a kontextustól függően sajátos, a köznyelvitől eltérő jelentéssel alkalmazza, és feltételezi, hogy ezt a jelentést a befogadó ismeri, de ha nem, annál jobb. (Sokal, Brickmont 1998; Bencze 1998)

1.2. Terminológiai szöveg

A terminológiai szöveg a szakszövegnek az a fajtája, amelyik *terminus technicusok* használatára alapoz. A terminus technicus az a szakkifejezés, amelynek értelmezésében bizonyos szakmai kör megegyezik, illetve amelyet a szövegalkotó következetesen, definiáltan alkalmaz. Szigorúan vett értelemben a definíció a terminus technicus szubsztanciális eleme. A szöveg befogadója előtt azonban nem jelenik meg mindig a definíció. Vagy azért, mert megállapodott szakszövegről van szó, ahol a terminusok jelentését normák, szótárak rögzítik, vagy azért, mert – főképp az újonnan bevezetett fogalmak esetében – a szövegalkotó első előfordulásakor értelmezvén az új fogalmat, a továbbiakban úgy tekinti, hogy a befogadó ezt megismerte. (Reformatszkij 1980)

Bár ez többnyire csak illúzió, a befogadót ez egyrészt ritkán zavarja, másrészt ritkán van belőle kára. Ráadásul érzékelhető, hol kell a szövegben terminusnak következnie, így azt a kifejezést, amelyet ott talál, eleve annak érzékeli, és ha a szöveget megérti, értelmezni is tudja (esetleg nem teljesen azonosan a szövegalkotó értelmezésével). Azt a helyet, ahol a befogadó terminust érzékel, terminológiai helyzetnek nevezzük, és az ilyen helyzetben megjelenő kifejezést pedig úgy tekintjük, hogy „eljátssza" a terminus szerepét.

1.3. Terminológiai helyzet

A terminológiai helyzet olyan kollokáció a szövegben, amelyet a befogadó különlegesnek érzékel, értelmezéséhez túl kell lépnie az adott kontextuson. Ezekben a helyzetekben terminus technicusnak kell megjelennie, amely a jelentését azzal a fogalmi rendszerrel összefüggésben kapja meg, amelyre vonatkoztatják. A terminológiai helyzet hiátusként jelenik meg a szövegben, amelyet terminus technicussal kell betölteni. Ennek az a következménye, hogy a terminológiai helyzetben megjelenő szót az olvasó terminus technikusnak érzékeli, elfogadja annak.

A terminológiai helyzet felismerésének két gyakorlati vonatkozását lehet megemlíteni: az egyik a reklám, amelyik arra a pszichológiai hatásra alapoz, hogy a vásárló bizalma nagyobb a „tudományosan igazolt" termékek iránt, a másik a szakfordítás, ahol a fordító felismeri, hogy egy kifejezés általa ismert jelentése nem illik az adott kontextusba.

1.4. Terminológiai szerep

Ha egy szó vagy kifejezés a terminológiai helyzetben jelenik meg, terminus technicusnak tűnik, függetlenül attól, hogy rendelkezik-e akár explicit, akár implicit definícióval. Ezt a jelenséget nevezhetjük *terminológiai szerepnek*. Ebben a helyzetben a lexéma másképp viselkedik, mint a köznyelvi helyzetben. Ekkor válnak rá jellemzőnek a terminológiával kapcsolatos specifikus megállapítások, például a terminológiai jelentés ideiglenes volta, a környezetfüggő környezetfüggetlenség. Ilyenkor válnak lehetségessé azok a terminológiai jellemzők, amelyek egyébként illuzórikusan fogalmazódnak meg a szakkönyvekben és szabványosítási előírásokban.

1.5. Monoszémia

A szakirodalom a terminus technicus preskriptív definíciójában a az egyik leglényegesebb követelménynek a monoszémiát tekinti, az egyalakúság mellett. A gyakorlat azt mutatja, hogy az egyalakúság (homonimamentesség) nem valósítható meg, és nem is okoz zavart a jelenléte, különösen, ha a szakág, amelyhez a terminológia tartozik, eléggé széles és tagolt. Ugyanakkor megfigyelhető, hogy az egyalakú szavak esetén az egyjelentésűség (monoszémia) általában érvényesül, tehát az azonos alakok jól elválasztható homonimák. A monoszémia nem úgy jelenik meg, hogy eltűnik a szóalak jelentésspektruma, hanem úgy, hogy a jelentéskör arra az egyetlen fogalomra fókuszál, amely a kommunikáció szakmai lényegéhez tartozik. Ez a szerep csak meghatározott helyzetben alakul ki. A dolog lényege az, hogy ugyanaz a szó a szövegben megjelenhet terminológiai helyzetben és azon kívül, ennek megfelelően játszhat terminológiai szerepet – esetleg többet is –, vagy köznyelvit.

1.6. A terminus technicus és a leírt fogalom meghatározása

A terminus technicus köznyelvtől örökölt jelentésspektrumát a terminológia klasszikus értelmezése szerint a definíció szűri, alakítja monoszémiává. A szakmai kommunikáció során azonban a definiálásra általában nincs idő. Ez azonban nem jelenti azt, hogy a terminológiai szerep és a terminológiai helyzet felismerését biztosító kompetencia valami másból táplálkozik. A terminológiai helyzetben megjelenő szavak szerepét valamiféle implicit definíció szabja meg. Ez az implicit definíció nem a szakszöveg, illetve a terminológia sajátossága. A múlt század 30-as éveiben megjelent szómezőelmélet szerint a szavak a jelentésük valamilyen közös ismérve alapján csoportokba foglalhatók. A közös ismérv mindenképpen implikálja a definíciót, tulajdonképpen ez a csoportosulás csak úgy képzelhető el, hogy a nyelvi kompetencia része a

fogalomelnevezés folyamatában a *genus proximumok* és a *differentia specificák* felismerése (pontos és teljes kijelölésük nélkül).

1.7. A terminológiai háló

Az, hogy egy-egy fogalmat képesek vagyunk vertikálisan és horizontálisan összekapcsolni más fogalmakkal, gondolkodásunknak hálós jelleget ad. A hálós struktúrát a függőleges hozzárendelések (egyedi fogalom a *genus proximumhoz*) és a vízszintes mellérendelések (egyedi fogalmak elválasztása a *differentia specificák* által) hozzák létre, melyben az egyes fogalmak a kapcsolódásaikkal jelentésmezőket alkotnak, és a mezők között is létrejön valamiféle hierarchikus rendezettség. Ha ennek topológiáját tudatosan leírjuk, lényegében megvalósítjuk a terminológiaalkotás klasszikus módszerét (a *genus proximumhoz* kötés és a *differentia specificák* megfogalmazása nem más, mint a klasszikus definiálás).

A tudatos terminológiaalkotás során ez a hálós struktúra szükségképpen létrejön. A tudatos terminológiaalkotás azonban ritka és nem is mindig hasznos tevékenység. A műszaki fejlődés felgyorsult, a szabványosítás nem tud lépést tartani az új eszközök megjelenésével, és így akár az alkalmazás kerékkötőjévé is válhat. Az előzőekben azonban utaltunk arra, hogy a szakszövegek fogalmi rendszerében megjelenik a definiáltság érzete, mondhatnánk, hogy ezek kvázidefiniált terminológiák.

A definíció, és az általa létrejövő egyértelműség elvileg meglehetősen határozott vonalú, kezelhető hálót tud alkotni. A gyakorlatban azonban ez az európai típusú szaknyelvek sajátja, amelyek ma nincsenek kifejezetten domináns helyzetben.

2. A számítógépes modell

A következőkben olyan, a szöveg felszíni jegyeire vonatkozó hipotéziseket írunk le, amelyet a projektünk elkövetkező szakaszaiban kell kísérletekkel igazolnunk. A kutatási feladat a következő:

Ismeretlen, de ismert tárgykörhöz tartozó forrásszövegben

(1) fel kell ismerni a terminológiai helyzetben levő (egy- és többszavas) lexémákat;

(2) a lexémák jelentésvektorában meg kell jelölni a terminológiai szerepet;

(3) a lexémát, pontosabban az adott jelentést el kell helyezni az 1.7. alatt említett terminológiai hálón, amennyiben lehetséges.

A harmadik feladat különösen akkor fontos, ha a gépi terminológiakivonatoló alkalmazás fordítási feladatokat támogat, hiszen ekkor nem csupán a terminológia „kijegyzetelése" a feladat, hanem – ideális esetben – a terminus technicusok fordításának meghatározása is. Erre a 4. részben visszatérünk.

A terminológiai helyzetet elsősorban két tényező határozhatja meg:

(1) Belső szerkezete: a terminus technicus jellemzően főnév vagy főnévi csoport, de ezen belül is meghatározott belső struktúrával rendelkezhet (a magyarban összetett, –i képzős jelzőket tartalmazhat; az angolban a főnévi jelzők halmozása vagy összetett birtokos szerkezet stb.). Igét, igei csoportot ritkán tekintünk terminus technicusnak,

ám a terminus technicussal egy igei csoportban szereplő ige meghatározhatja a kérdéses lexéma terminológiai szerepét.

(2) A szövegben elfoglalt helye, közeli és távoli kollokációi: bizonyos tárgykörökben egyes jelzők, határozószók vagy igék terminológiai helyzetet indukálhatnak. Ezek jellemzőit nyelv- és témafüggő módon, korpuszkutatással, a kontextus statisztikai vizsgálatával becsülhetjük. A távolabbi kontextusban pedig olyan jegyeket vizsgálhatunk, mint

a) főnévi csoport előfordulása címben vagy definitív szerkezetű mondatban;

b) főnévi csoport első előfordulása a szövegben, hangsúlyos helyzetben.

A statisztikai vizsgálatok elsősorban azt tudják kimutatni, hogy egy kollokáció tagjai mutatnak-e speciális affinitást egymás irányában. Ezzel részletesen foglalkozik a számítógépes terminológia irodalma (Jacquemin 2001), de magunk is írtunk róla, többek között az előző évi MSZNY-konferencián (Kis-Ugray 2003; Kis et al 2004.). A korpuszstatisztika alkalmas arra, hogy felismerje a többszavas lexémák (esetünkben a többszavas terminus technicusok) elemeinek összetartozását, illetve jellemezze a terminusjelöltek környezetét. Az alaphipotézis itt mindig az, hogy a vizsgált elemek együtt nagyobb valószínűséggel fordulnak elő, mint egymástól függetlenül.

Példa adott lexéma különböző terminológiai helyzetére:

> The *key* tool you'll use to manage system *processes* and applications... (key = legfontosabb, nincs terminológiai helyzetben)
>
> Provides protected storage for sensitive data, such as private *keys*... (key = kulcs, titkosítási kulcs)
>
> [William R. Stanek (2000): Microsoft Windows 2000 Administrator's Pocket Consultant. Microsoft Press (Redmond).]

A másik feladat a terminusjelölt – a vizsgálat központjában álló lexéma – terminológiai szerepének kijelölése. Ez elsősorban egyértelműsítési probléma, hiszen a lexéma jelentésvektorából kell egy jelentést kiválasztani. Mivel a terminológiakezelés elsősorban az emberi fordítást támogatja, a jelentést sok esetben elegendő a lexéma célnyelvi fordításával reprezentálni (amennyiben rendelkezésre áll). A probléma a szójelentés-egyértelműsítés (WSD) speciális esetének is tekinthető, amelynek során nyelv- és tárgykörfüggő módon járunk el. A terminológiai szerep kiválasztása jelentheti a poliszém terminus egyértelműsítését is. Példa:

> (1) a group of computers that share a common directory database.
> (számítógépek csoportja, amely közös címtáradatbázist használ)
>
> (2) To rename a file or directory, follow these steps:
> (Fájlok vagy könyvtárak átnevezéséhez végezzük el az alábbi lépéseket:)
>
> [William R. Stanek (2000): Microsoft Windows 2000 Administrator's Pocket Consultant. Microsoft Press (Redmond).]

Azonban az egyértelműsítés vezethet oda is, hogy a kijelölt szerep szerint a terminusjelölt kikerül a terminológiai helyzetből. Ez természetesen lehetséges, hiszen a terminológiai helyzet megállapítása a rendelkezésünkre álló eszközökkel – még a fenti

megközelítésben sem – érheti el a 100%-os pontosságot. Így minden, terminológiai helyzetben levőnek megjelölt lexéma (terminusjelölt) terminológiai helyzetéhez hozzárendelhető egy konfidenciaérték (pontszám), amely jellemzi a terminológiai helyzet megállapításának „jóságát" az adott helyen.

A fentiekből következik, hogy a terminológiai helyzet és a terminológiai szerep nem független változók, az egyik vizsgálata kihat a másikéra és viszont. Mindkettő a kollokációelemzés valamilyen alkalmazása.

Még egy fontos megjegyzést kell tennünk. Bár a terminológiakivonatolási feladat új, ismeretlen forrásnyelvi szöveg feldolgozását követeli meg, nincs szó arról, hogy az adott tárgykör (domain) forrásnyelvi terminológiája egészében ismeretlen volna. Feltételezhetjük, hogy a művelet kezdetén rendelkezésre áll a forrásnyelvi terminusok egy zárt halmaza, s ezt felhasználhatjuk a terminológiakivonatolás során. Ez egyfelől megkönnyíti a terminológiai helyzetek felismerését: erős jelöltek a halmazban már szereplő lexémák és közvetlen, a lokális szintaktikai feltételeknek megfelelő kollokációik. Másfelől lehetővé teszi a terminológia induktív jellegű feltárását. (Chien-Chen 2001)

A vázolt terminológiakivonatolási modellhez szükséges számítógépes nyelvészeti apparátus a következő:

(1) Morfológiai és kis mélységű, részleges szintaktikai elemzés: ez elsősorban a közeli (közvetlen) kollokációk szerkezetének vizsgálatára való; a rendszerben ugyanis meg kell határozni a forrásnyelvi terminus technicusok lehetséges szintaktikai szerkezetét, s ezek alapján lehet minősíteni a jelölteket. A szabályokat ideálisan korpuszelemzéssel, öntanuló mechanizmusokkal alakítjuk ki

(2) Speciális, az egymástól távol levő elemek közötti összefüggéseket feltáró elemző: ez a lokálisan feltárt jelöltek és a fejezetcímek, illetve az egy szövegegységbe eső jelöltek közötti kapcsolatok feltárására való

(3) Kollokációkeresési (kollokációstatisztikai) infrastruktúra

(4) Kiindulási lexikon

A számítógépes terminológia irodalma megkülönböztet statisztikai és szabályalapú (szótáras, illetve sekély nyelvtani elemzésre épülő) terminológiakivonatolási rendszereket (Jacquemin 2001; Castellví et al. 2001). Az itt vázolt megközelítés abban jelent újdonságot, hogy koherens, deskriptív definíciót keres a terminológiára, s ez alapján, rendszerszemléletben építi fel a terminológiakivonatolási folyamatot. Emellett gyakorlati különbség jelentkezik a nyelvészeti erőforrások felhasználásában is.

3. A terminológiakivonatolás nyelvi erőforrásai

3.1. Korpuszok és szószedetek

A terminológiakivonatoló rendszernek – sőt, semmilyen nyelvtechnológiai rendszernek – nem szabad elveszítenie a kapcsolatot az élő nyelvvel. Ha egy alkalmazás egy szinkrón korpusz egy adott állapota alapján készül, s a nyelvi erőforrást ezt követően statikusan kezeli, elavul, mert képtelen a nyelv változásait követni. Gyorsan változó szakterületek esetén új terminusok, új kontextusok akár havonta megjelenhetnek; ugyanakkor nem állíthatjuk azt, hogy a terminusok szintaktikai szerkezete hasonló

ütemben változna. Ezzel együtt annak változása sem zárható ki, épp az ad hoc, metaforikus és intuitív terminológiaalkotás miatt.

Az itt javasolt terminológiakivonatolási séma alapja ideális esetben egy hálózati szolgáltatás. Központi helyen gyűjtjük meghatározott nyelv(ek) és az(ok)on belül szakterületek korpuszait és szószedeteit, s ezek felhasználásával rendszeresen újrageneráljuk a kivonatolási szabályokat. E nyelvi erőforrások rendelkezésre állása esetleges, a rendszernek pedig lehetőleg minden körülmények között törekednie kell használható kimenet előállítására. Ezért a szabályokat a következő esetek mindegyikére ki kell dolgozni:

- rendelkezünk a célnyelven és a forrásnyelven szaknyelvi korpusszal és kétnyelvű szótárral,
- rendelkezünk kétnyelvű szótárral, de csak a forrásnyelven elérhető a szaknyelvi korpusz,
- nem rendelkezünk kétnyelvű szótárral, de rendelkezünk cél- és forrásnyelvi szaknyelvi korpusszal,
- nem rendelkezünk sem kétnyelvű szótárral, sem célnyelvi szaknyelvi korpusszal.

3.2. A terminológiai háló felhasználása

Ha egy forrásnyelv–szakterület pár terminológiáját hálóba szervezzük, amelyben fel lehet tüntetni a szakterület részterületeit (legalább ezeket), illetve a szakterülethez tartozó genus proximumokat, a terminológiai szerep felismerésében – a terminusjelöltek egyértelműsítésében – felhasználható a terminusjelölt fogalmi kerete. Ehhez meg kell határoznunk, hogy a terminusjelölt környezetében milyen más megerősített terminusok, illetve terminusjelöltek helyezkednek el. Példa a *directory* szó két jelentésének fogalmi kereteire:

directory (címtár)
(1) Active, database, service
(2) domain, forest, operations master replication, site, tree, user
(3) domain, forest, operations master replication, site, tree, user
directory (könyvtár)
(1) name, sub+
(2) copy, delete, file, folder, move, rename, select
(3) copy, delete, file, folder, move, rename, select

4. A projekt és alkalmazása

A projekt célja fordítástámogató eszköz kifejlesztése, amely automatikusan és a lehető legnagyobb pontossággal kigyűjti az új forrásszövegekből a terminus technicusokat, és amennyiben a kérdéses nyelvpárhoz rendelkezésre áll szószedet, ezek célnyelvi megfelelőit is megadja. A robusztus és megbízható működés azért fontos, mert a fordítástámogatás lényege a fordítási munka felgyorsítása, hatékonyságának növelése. A fordítók, fordítási projektek számára a terminológia meghatározásában nem jelenthet több-

letmunkát egy ilyen eszköz alkalmazása. Ezért nagyon fontos a megfelelő értékelési, kiértékelési munka a fejlesztés során.

A fejlesztés során angol és magyar nyelvű szövegek feldolgozására, kivonatolására készítjük fel a rendszert. Emellett mindkét forrásnyelvhez kiválasztunk legalább három, az EU-csatlakozáshoz szorosan kapcsolódó témakört: az alapadatokat először ezekhez állítjuk elő. Ezzel azt a stratégiai célt is szolgáljuk, hogy a magyar terminológia minél előbb és minél nagyobb mértékben kapcsolódjon az EU szabványos terminológiájához, a fordítószolgálatok által is használt szabványos terminológiai adatbázisokhoz.

Köszönetnyilvánítás

Az itt vázolt terminológiakivonatolási séma fejlesztése az IKTA-5-181/2003. sz. projekt keretében zajlik, amelynek jelenleg a 2. munkaszakaszánál – a referenciakorpuszok és -szószedetek összegyűjtésénél – tartunk.

Irodalom

BENCZE Gyula (1998): Posztmodern panoptikum Magyar Tudomány, 1998. december, (http://www.kfki.hu/~cheminfo/hun/teazo/sokal/panop.html).

CASTELLVÍ, M. Teresa Cabré – BAGOT, Rosa Estopà – PALATRESI, Jordi Vivaldi: Automatic Term Detection: A Review of Current Systems. In: BOURIGAULT, Didier – JACQUEMIN, Christian – L'HOMME, Marie-Claude (eds.): Recent Advances in Computational Terminology. John Benjamins, Amsterdam-Philadelphia, 2001. pp. 53–88.

CHIEN, Lee-Feng – CHEN, Chun-Liang (2001): Incremental Extraction of Domain-specific Terms from Online Text Resources. In: BOURIGAULT, Didier – JACQUEMIN, Christian – L'HOMME, Marie-Claude (eds.): Recent Advances in Computational Terminology. John Benjamins, Amsterdam-Philadelphia, pp. 89-111.

JACQUEMIN, Christian (2001). Spotting and Discovering Terms through Natural Language Processing. The MIT Press.

KIS Balázs–UGRAY Gábor (2003): Új korpuszstatisztikai eszköztár kollokációkeresésre. In: Az I. Magyar Számítógépes Nyelvészeti Konferencia gyűjteményes kötete, Szegedi Tudományegyetem, Szeged.

KIS, Balázs–VILLADA MOIRÓN, Begoña–BÍRÓ, Tamás–BOUMA, Gosse–NERBONNE, John–POHL, Gábor–UGRAY, Gábor (2004): A New Approach to the Corpus-based Statistical Investigation of Hungarian Multi-word Lexemes. In: Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC-2004), Lisbon, Portugal.

REFORMATSZKIJ, A.A. (1980): Miszli o tyerminologii. in in: Danilenko V.P. red.: Szovremennije problemi russzkoj tyerminologii, NAuka, Moszkva, p 163.

SOKAL, ALAN és BRICKMONT, Jean (1998): Mi ez a nagy cirkusz? Magyar Tudomány,1998. április

III. Fordítás

GeLexi projekt: Gépi fordítás totálisan lexikalista alapokon

Alberti Gábor, Kleiber Judit, Viszket Anita*

Pécsi Tudományegyetem Bölcsészettudományi Kar Nyelvtudományi Tanszék
7624 Pécs, Ifjúság útja 6., gelexi@btk.pte.hu

Abstract. Cikkünkben a gépi fordítás egy új (totálisan lexikalista) megközelítését mutatjuk be, miután röviden ismertettük a GASG implementációján alapuló elemzőnket, amelynek kétirányú alkalmazásával (egy speciális, ún. kopredikációs hálózaton keresztül) valósítható meg a fordítás. Programunk újdonsága többek között abban rejlik, hogy a szintaktikai mellett szemantikai reprezentációt is képes társítani a mondatokhoz (ezáltal géppel segített fordításként is felfogható); továbbá hogy nem nyelv-specifikus, ezért bármely nyelvről képes bármely másikra fordítani, ha azok lexikai egységeit (minden tulajdonságukkal együtt) az elemző tartalmazza. Eddig elért eredményeinket (magyar és angol mondatok egy körének elemzése és fordítása) példákon keresztül is szemléltetjük.

1 Bevezetés

Kutatócsoportunk alapvető célja annak igazolása, hogy a számítógépes nyelvészetnek [13] érdemes visszafordulnia a tiszta (generatív) nyelvelméleti alapok felé. Ezt egyrészt a technológia fejlődése teszi lehetővé, másrészt a nyelvtudományban megfigyelhető erőteljesen lexikalista fordulat, a morfológia és a szintaxis egységes kezelése és a dinamikus (diskurzus)szemantikai elméletek előtérbe kerülése.

A gépi fordítás itt bemutatott megközelítését egy Karttunen radikális lexikalizmusát [13] továbbfejlesztő totálisan lexikalista nyelvtan, a GASG (*Generative / Generalized Argument Structure Grammar* [2] [7]) implementálása kapcsán dolgoztuk ki. A totális lexikalizmus azt jelenti, hogy minden információt a lexikonban tárolunk, nincs szükség frázisstruktúra építésére. A GASG egy módosított unifikációs kategoriális nyelvtannak tekinthető, amely már az egyetlen szintaktikai műveletet (a függvényalkalmazást) sem tartalmazza. Ami marad, az csupán a lexikai egységek maguk nagyon gazdag belső szerkezettel, és az unifikáció, mint az elemek kombinálhatóságáért felelős művelet.

A nyelvtan fontos tulajdonsága továbbá, hogy – mivel nem épít frázisstruktúrát – mozgatást sem tartalmaz, a szórendről mégis számot ad, mégpedig ugyanúgy unifikáció segítségével, mint az esetről vagy az egyeztetésről. A lexikai egységek

* A cikk megírását és a szegedi konferencián való jelenlétünket a T 38386 számú OTKA pályázat tette lehetővé.

leírásában rögzítjük a saját tulajdonságaikat, illetve azokat az elvárásokat, amelyeket a velük kapcsolatot létesítő lexikai egységektől elvárnak. Egy ilyen követelmény az is, hogy mely elemekkel akarnak szomszédosak lenni, és ez az igény mennyire erős rangú, hiszen más egységek is akarhatnak az adott elem mellett lenni a mondatban. További újdonsága a GASG-nek, hogy a lexikai egységek nem kész szavak, hanem (szabad vagy kötött) morfémák (tövek és toldalékok), ők létesítik a szintaktikai viszonyokat, és belőlük számítódik ki a mondat szemantikája. Végül fontosnak tartjuk kiemelni, hogy a nyelvtanhoz (így az implementációhoz is) szemantikai komponens is tartozik, ahol a mondat jelentése a lexikai egységek összeépülése során közvetlenül előáll. A reprezentációt pedig egy dinamikus diskurzus-szemantikai keretben (ReALIS [4]) képzeljük el, amely a DRT [11] egy továbbfejlesztett változata.

A nyelvtanhoz tartozó implementációt 2001 óta fejlesztjük, mindig új jelenségekkel bővítve. A tavalyi konferencián (MSZNY2003) a főnévi csoportok közötti *és* kötőszóról beszéltünk [9], idén a vonzatos melléknév kezelését mutatnánk be fő témánk, a gépi fordítás totálisan lexikalista megközelítése mellett.

2 Az elemző

A Prológban írt elemzőnk bemenete egy (magyar vagy angol nyelvű) szósor, amiről a program eldönti, hogy jól formált szavakból áll-e, grammatikus mondatot alkot-e, és társítható-e hozzá jól formált diskurzus-szemantikai reprezentáció. Ha mindez teljesül, kimenetként olvashatjuk a mondatot alkotó szabad és kötött morfémák (tövek és hozzájuk tartozó toldalékok) listáját, a morfémaszinten lévő lexikai egységek közötti grammatikai viszonyokat (egy alaposabb és egy áttekinthetőbb formában), a DRS-t (diskurzusreprezentációs struktúrát), amit a mondathoz társíthatunk, illetve a szintaxis és a szemantika között elhelyezkedő kopredikációs hálózatot, amely a fordítás során bizonyul hasznosnak.

A program működését egy vonzatos melléknevet tartalmazó mondat magyar és angol nyelvű megfelelőjének elemzésén keresztül világítjuk meg.

```
gramm("Egy Marira büszke fiú énekeltetheti Julit.").
LEXIKAI EGYSÉGEK:
...
énekel: n(5,1,li(m("","énekel",""),labstem("sing",phonfst(1,2,2,2)...)))
tet: n(5,2,li(m("t","A","t"),labder("cause",phonfsu...)))
het: n(5,3,li(m("h","A","t"),labsuff("may",phonfsu(1,1,1,2),2,1)))
i: n(5,4,li(m("","i",""),labsuff("sg3obj+def",phonfsu(1,3,1,3),2,3)))
...
SZINTAXIS:
...
gr("noun","regent","subj",4,1,5,1)
gr("regent","noun","subj",5,1,4,1)
gr("regent","det","subj",5,1,1,1)
gr("suff","stem","free",5,2,5,1)
...
det-noun: egy-fiú
adj-noun: büszke-fiú
regent-noun-obl: büszke-Marira
regent-det-obl: büszke-Marira
regent-noun-subj: énekeltetheti-fiú
regent-det-subj: énekeltetheti-egy
```

```
regent-noun-obj: énekeltetheti-Julit
regent-det-obj: énekeltetheti-Julit
SZEMANTIKA:
provref("fixpoint",[e(5,3,1)])
provref("new",[r(1,1,1)])
provref("<or=",[r(1,1,1),e(5,2,1)])
provref("old",[r(2,1,1)])
pred("Mary",2,[r(2,1,1)])
pred("proud",3,[r(1,1,1),r(2,1,1)])
pred("boy",4,[r(1,1,1)])
provref("new",[e(5,1,1)])
pred("sing",5,[e(5,1,1),r(6,1,1)])
provref("new",[e(5,2,1)])
provref("=",[e(5,2,1),e(5,1,1)])
pred("cause",5,[e(5,2,1),r(1,1,1),e(5,1,1)])
provref("new",[e(5,3,1)])
provref("<",[e(5,3,1),e(5,2,1)])
pred("may",5,[e(5,3,1),e(5,2,1)])
provref("old",[r(6,1,1)])
pred("Julie",6,[r(6,1,1)])
yes
```

Az elemzés végén a *yes* jelenti, hogy a mondat grammatikus, előtte pedig a különböző kimeneteket olvashatjuk. Először a lexikai egységek sorozatát, amelyekből itt az *énekeltetheti* szót alkotó morfémákat emeltük ki. A sorok elején találhatók az adott allomorfok, majd a lexikai egység a mondatbeli számával (hányadik szó hányadik morfémája), a változókat tartalmazó sajáttestével (amely azt mutatja, hogy milyen alakokban jelenhet meg), majd pedig egy címkével, amelyben a különböző (fonológiai, morfológiai, szintaktikai stb.) tulajdonságai vannak tárolva.

Az első (részletesebb) szintaktikai reprezentáció megmutatja, hogy melyik lexikai egység (első két szám) melyik másikkal (második két szám) pontosan milyen kapcsolatot létesít. A viszonyok lehetnek egyirányú, szabad ("free") viszonyok (pl. melléknév viszonya a főnévhez, szuffixumé a tőhöz), illetve kölcsönös vonzat-viszonyok ("subj", "obj" stb.), amelyet mindig két pilléren (főnévi és determinánsi) kell megtalálni. A második (egyszerűbb) szintaktikai reprezentációban a viszonyokat könnyebben áttekinthető formában olvashatjuk.

A szemantikai reprezentáció még a korábban használt elméleti keretet, az LDRT-t [3] tükrözi, de hamarosan áttérünk a ReALIS interpretálására, amely jóval precízebb elemzést tesz lehetővé. A bevezetett referensekről különféle állításokat teszünk, így kapjuk meg a mondat jelentését. Példánkban a fő állítás (a fixpont) az e531 állítás, hogy lehetséges egy e521 szituáció, azaz, hogy r111 (egy fiú, akiről tudjuk, hogy büszke Marira) okoz egy e511 eseményt, vagyis azt, hogy r611 (Juli) énekel.

Érdekes nyelvészeti problémákat vet fel az az eset, amikor a melléknév vonzata (a bővített köznévvel megegyező határozottságú) köznév (pl. *a fiúra büszke lány*), amikor is a magyarban névelőtörlés történik. A probléma kezelését határozottsági rangparaméterekkel [5] oldjuk meg, és az előadásunkban mutatnánk be részletesebben.

Programunk angol mondatokat is tud elemezni, ennek szemléltetésére álljon itt az előző példa.

```
gramme("A boy proud of Mary may make Julie sing.").
LEXICAL ITEMS:
...
proud: n(3,1,li(m("","proud",""),labsteme("proud",4,[["of"]])))
...
SYNTAX:
...
det-noun: a-boy
adj-noun: proud-boy
regent-noun-obl: proud-Mary
regent-det-obl: proud-Mary
regent-verb-arg: may-make
regent-noun-subj: make-boy
regent-det-subj: make-a
regent-verb-arg: make-sing
regent-noun-subj: sing-Julie
regent-det-subj: sing-Julie
SEMANTICS:
provref("fixpoint",[e(6,1,1)])
provref("new",[r(1,1,1)])
provref("<or=",[r(1,1,1),e(7,1,1)])
pred("boy",2,[r(1,1,1)])
pred("proud",3,[r(1,1,1),r(5,1,1)])
provref("old",[r(5,1,1)])
pred("Mary",5,[r(5,1,1)])
provref("new",[e(6,1,1)])
provref("<",[e(6,1,1),e(7,1,1)])
pred("may",6,[e(6,1,1),e(7,1,1)])
provref("new",[e(7,1,1)])
provref("=",[e(7,1,1),e(9,1,1)])
pred("cause",7,[e(7,1,1),r(1,1,1),e(9,1,1)])
provref("old",[r(8,1,1)])
pred("Julie",8,[r(8,1,1)])
provref("new",[e(9,1,1)])
pred("sing",9,[e(9,1,1),r(8,1,1)])
yes
```

Láthatjuk, hogy a lényeges kimenet (a szemantika) valójában ugyanaz, mint a magyar példánál, csak a számok mások, hiszen más sorrendben következnek a szavak az angol mondatban, mint a magyarban, és a szószint sem ugyanott húzódik a két nyelv esetében.

3 Géppel segített fordítás

Az imént bemutatott különböző forrásnyelvi mondatok DRS-ei közötti egyezés lehetőséget ad arra, hogy a szemantikai elemzőnket a géppel segített fordítás területén hasznosítsuk. Hiszen míg egy idegen nyelv megtanulása évekig eltart, a DRS-ek olvasását néhány óra alatt el tudja sajátítani az angolul tudó beszélő[1].

Feladatunk tehát minél több nyelvre kidolgozni az elemzőt, így minél több nyelv esetében tudjuk megvalósítani a géppel segített fordítást azáltal, hogy univerzális szemantikai elemzőnk lényegében angol DRS-eket hoz létre. Egy ilyen DRS az angol nyelv egyértelműsített, formalizált (így megszorított) változatának tekinthető, ahol még az amúgy (legtöbbször) implicit idő- és térreferensek is megjelennek.

[1] A DRS-ek "nyelve" jelenleg az angol, de ez csupán egy praktikus választás volt, bármely más nyelv megfelelő lenne.

Különösen nagy nehézséget okozhat egy idegen anyanyelvűnek az agglutinatív nyelvek megértése, amilyen például a magyar is. Ennek oka, hogy például a szereplők személyére sokszor csak a ragozásból lehet következtetni, amikor a névmások hiányoznak. Elemzőnk természetesen felfejti ezeket a viszonyokat, és szerepelteti a szemantikai reprezentációban, így nem okozhat nehézséget a mondat megértése. A következő egyszerű példából láthatjuk mindezt: világosan kiderül a DRS-ből, hogy a *szeret* predikátum első argumentuma (alanya) az r011 referens, ami az én jelölésére alkalmas (a 0 jelenti, hogy beépített referensről van szó, az 11 pedig, hogy egyes szám első személyű), a második argumentuma (tárgya) pedig az r012, vagyis te (12: egyes szám második személy).

```
gramm("Szeretlek.").
LEXIKAI EGYSÉGEK:
szeret:
n(1,1,li(m("","szeret",""),labstem("love",phonfst(1,2,2,2),2...)))
l: n(1,2,li(m("","l",""),labsuff("objperson2",phonfsu(3,2,1,1),2,2.5)))
ek: n(1,3,li(m("V","k",""),labsuff("sg1",phonfsu(1,1,2,3),2,3)))
SZINTAXIS:
gr("suff","stem","free",1,2,1,1)
gr("suff","stem","free",1,3,1,1)
SZEMANTIKA:
provref("fixpoint",[e(1,1,1)])
provref("new",[e(1,1,1)])
pred("love",1,[e(1,1,1),r(0,1,1),r(0,1,2)])
```

Végül megemlítjük, hogy a szemantikai kimenetet valamilyen adatbázisban is eltárolhatjuk, ami alapján különféle lekérdezéseket végezhetünk, így könnyítve meg még jobban a DRS-ből információt kinyerni szándékozó felhasználó dolgát.

4 Gépi fordítás

A gépi fordításhoz elemzőnk kétirányú használatával juthatunk el: a program először ellenőrzi a forrásnyelvi mondat grammatikalitását és előállítja a különböző kimeneteket, majd generálja a célnyelvi mondatot ezen kimenetek alapján. Amit felhasznál a generáláshoz, az a predikáló lexikai egységek sorozata (nem használja például az egyeztetésért felelős morfémákat, ha az adott információt nem csupán ők hordozzák), és a korábban érintőlegesen megemlített kopredikációs hálózat.

Felmerülhet a kérdés, hogy miért nem egyszerűen a szemantikai reprezentációt használjuk a fordításhoz, miért van szükség egy újabb szint közbeiktatására. A válasz abban rejlik, hogy a forrásnyelvi mondat minél hűbb fordítását szeretnénk előállítani, és lehetnek olyan információk, amelyek a DRS-ben már nem szerepelnek, mert nem tartoznak szigorúan a mondat jelentéséhez, inkább a formájáról árulkodnak. A kopredikációs hálózat pedig egy olyan szint, amely a szintaxis és a szemantika között helyezkedik el: még őriz valamennyit a forrásnyelv eredeti struktúrájából, de már mutatja a szemantikai viszonyokat. A fordításra az teszi alkalmassá, hogy az egyes mondatok ábrázolása már ezen a szinten sem mutat különbséget még olyan nyelvek között sem, ahol tradicionálisan nagy különbségeket szoktak feltételezni, mint például a magyar és az angol. Ezt kívánjuk szemléltetni a két korábban bemutatott mondaton: *Egy Marira büszke fiú*

énekeltetheti Julit, illetve *A boy proud of Mary may make Julie sing.* A két mondat formája között alapvető különbségek vannak (például máshol található a szószint), a kopredikációs hálózataik azonban (a morfémák számozásától eltekintve) tökéletesen megegyeznek.

```
KOPREDIKÁCIÓS VISZONYOK:
copr("a(n)"..."boy"...0,1,"free")
copr("proud"..."boy"...1,1,"free")
copr("proud"..."Mary"...2,1,"arg")
copr("proud"..."Mary"...2,0,"arg")
copr("sing"..."Julie"...1,1,"arg")
copr("sing"..."Julie"...1,0,"arg")
copr("cause"..."boy"...1,1,"arg")
copr("cause"..."a(n)"...1,0,"arg")
copr("cause"..."sing"...2,0,"arg")
copr("may"..."cause"...1,0,"arg")
```

```
COPREDICATIVE NETWORK:
copr("a(n)"..."boy"...0,1,"free")
copr("proud"..."boy"...1,1,"free")
copr("proud"..."Mary"...2,1,"arg")
copr("proud"..."Mary"...2,0,"arg")
copr("may"..."cause"...1,0,"arg")
copr("cause"..."boy"...1,1,"arg")
copr("cause"..."a(n)"...1,0,"arg")
copr("cause"..."sing"...2,0,"arg")
copr("sing"..."Julie"...1,1,"arg")
copr("sing"..."Julie"...1,0,"arg")
```

Az első hálózat tartozik a magyar, a második az angol nyelvű mondathoz. Látható, hogy melyik két predikátum kopredikál (a predikátum neve után a száma szerepelne), illetve, hogy azok melyik argumentuma (első, második vagy nulladik, azaz szituációs), és, hogy szabad- vagy vonzatviszonnyal. Például a *cause (okoz)* predikátum három másik elemmel kopredikál, a *fiú* predikátummal, ami az első argumentuma, az *a(n) (egy)* predikátummal, ami az első argumentumának determináló pillére, és a *sing (énekel)* predikátummal, aminek a nulladik argumentuma (az éneklés maga) az ő második argumentuma.

A fordításhoz természetesen nem elég, hogy az elemzés ugyanarra az eredményre vezet ugyanolyan tartalmú, de különböző nyelvű mondatok esetében. A célnyelvi mondatot elő kell állítani (generálni kell), helyes szórenddel és megfelelő egyeztetéssel. Azt is tudni kell továbbá, hogy milyen argumentumszerkezettel kell szerepeltetni egy adott régenst, hiszen e téren sincs egyértelmű megfeleltetés a különböző nyelvek azonos funkciójú szerkezetei között. A modellünk és az implementációnk megoldást tud nyújtani mindezekre a problémákra.

Az egyeztetésről úgy tudunk számot adni, hogy a generálás során nem csupán a célnyelvi predikáló lexikai egységek forrásnyelvi megfelelőit gyűjtjük össze, hanem a nyelvspecifikus, egyeztetésért felelős morfémákat is. Ezeket változók formájában keressük, típusaik és pozícióik univerzálisak [8], körük egy adott nyelv tekintetében pedig még tovább szűkíthető. A célnyelvi elemző (grammatikalitás-ellenőrző) pedig kiszűri a hibás alakokat, csak a helyeset hagyva meg. A processzálási idő csökkentése érdekében természetesen egyéb szűrők is beépíthetők a programba. A helyes szórend kialakításáért a már említett szomszédossági rangparaméterek felelősek. A generálás első lépése (a lexikai egységek összegyűjtése) után azok minden lehetséges variációját előállítjuk, majd kiszűrjük a triviálisan lehetetlen változatokat, végül a célnyelvi elemzőbe épített megelőzési ragparaméterek csak a helyes szórendű mondatot találják grammatikusnak, így az lesz a forrásnyelvi mondat fordítása. Az argumentumstruktúra (esetjelölő morfémák) kiszámítására pedig létezik egy ágentív hierarchián alapuló kalkulációs eljárás [1] [8], amit itt nem tárgyalunk részletesebben.

Végül tekintsük meg, milyen választ ad a programunk, ha a korábbi két mondat fordítására vagyunk kíváncsiak, illetve, ha egy adott mondat esetében az összes lehetséges fordítást látni szeretnénk (ezt Prológban a *fail* paranccsal

érhetjük el).

```
translate_Hun_Eng("Egy Marira büszke fiú énekeltetheti Julit.").
In English: A boy proud of Mary may make Julie sing.
yes

translate_Eng_Hun("A boy proud of Mary may make Julie sing.").
In Hungarian: Egy Marira büszke fiú énekeltetheti Julit.
yes

translate_Eng_Hun("I love you."),fail.
In Hungarian: Szeretlek.
In Hungarian: Szeretlek téged.
In Hungarian: Szeretlek titeket.
In Hungarian: Én szeretlek.
In Hungarian: Én szeretlek téged.
In Hungarian: Én szeretlek titeket.
no
```

5 Összegzés

Alapvető célunk tehát legitimálni egy új, totálisan lexikalista nyelvtant azáltal, hogy a számítógépes implementálhatóságát igazoljuk, hiszen ez a legjobb módszer egy formális rendszer egzaktságának és konzisztenciájának a bizonyítására. Azon dolgozunk, hogy egyelőre a magyar és az angol, később egyéb nyelvek elemzőit elkészítsük, úgy, hogy ezek az elemzők képesek legyenek a generatív alapfeladat végrehajtására: ellenőrizni a szavak és a mondatok jólformáltságát, és szemantikai reprezentációt társítani szövegekhez. További célunk a cikkben bemutatott új megközelítés segítségével minél tökéletesebb gépi fordítást végezni, elemzőnk kétirányú használatával. Jelenleg egyre növekvő szókészleten egyre több nyelvészeti jelenséget tartalmazó mondat elemzésére vagyunk képesek, és ezeket a mondatokat fordítani is tudjuk angolról magyar nyelvre és fordítva.

Programunk több téren is újdonságot nyújt. A legfontosabb, hogy működő szemantikai komponenst tartalmaz, és ténylegesen képes a beírt mondatokhoz modern szemantikai reprezentációt társítani. A megközelítésünk alapjául szolgáló totális lexikalizmusnak elméleti és gyakorlati előnyei is vannak. Elméleti előny a homogenitás, azaz nincs külön szintaxis és lexikon, csak lexikon van[2]. Elméleti és gyakorlati előny egyben maga a lexikalizmus, amellyel a szabadabb szórendű nyelvek (mint a magyar) könnyebben kezelhetők. Programunk tisztán gyakorlati előnye pedig a számítástechnikában kívánatos "minimális processzálás – maximális adattár" [6]. Végül mi az előnye a fordítás totálisan lexikalista megközelítésének? Az, hogy univerzális tud lenni a keret, amelyet használ, nem pedig nyelvspecifikus, ezért nem kell külön külön kidolgozni minden egyes nyelvpárra a fordítás mechanizmusát. Amint a nyelvek elemzői rendelkezésünkre állnak, bármely nyelvről fordítani tudunk a másikra. Ezért is van, hogy egyidejűleg

[2] Egyéb homogén rendszerek is léteznek, amelyek azonban inkább a lexikont száműzik, és csak szintaktikai szabályokkal dolgoznak (pl. [14]). Azonban a nyelvészetben az utóbbi években megfigyelhető erősen lexikalista fordulat inkább a mi megközelítésünket igazolja, legalábbis elméleti szemszögből.

működik programunkban a magyarról angol nyelvre, illetve az angolról magyar nyelvre történő fordítás.

A továbbiakban szeretnénk az elemzett jelenségek körét egyre jobban kiterjeszteni, hogy egyre bonyolultabb (magyar és angol nyelvű) mondatokat legyünk képesek kezelni. Készül továbbá egy nyelvészeti adatbázis (LiLe projekt [10]), amelyet alapul véve rendszerünk jóval több szót lesz képes felismerni. Szeretnénk továbbá szövegek elemzésére képessé tenni a programunkat, hiszen a szemantikai reprezentáció erre lehetőséget ad. Végül pedig célunk más nyelvek nyelvtanainak kidolgozása, hogy így a (nem nyelv-specifikus) fordítási mechanizmusunkat minél több nyelvre alkalmazhassuk.

References

1. Alberti, Gábor (1997): Argument Selection. Peter Lang, Frankfurt am Main
2. Alberti, Gábor (1998): GASG: Minimal Syntax, Maximal Lexicon and PROLOG, paper read at ALLC/ACH '98, July 9. In Hunyadi, L. (ed.): ALLC/ACH '98. KLTE, Debrecen. 81-83
3. Alberti, Gábor (2000): Lifelong Discourse Representation Structures, Gothenburg Papers in Computational Linguistics 00 5. 13-20
4. Alberti, Gábor (2004): ReAL Interpretation System. In L. Hunyadi, Gy. Rákosi, E. Tóth (eds.): The Eighth Symposium on Logic and Language, Preliminary Papers. 1-12
5. Alberti Gábor, Balogh Kata (2003): Az eltűnt névelő nyomában. Megj. előtt Büky L. (szerk.): A mai magyar nyelv leírásának újabb módszerei VI. SZTE, Szeged. 9-31
6. Alberti Gábor, Balogh Kata, Kleiber Judit, Viszket Anita (2002): A totális lexikalizmus elve és a GASG nyelvtan-modell. Maleczki M. (szerk.): A mai magyar nyelv leírásának újabb módszerei V. Szegedi Tudományegyetem. 193-218
7. Alberti Gábor, Kata Balogh, Judit Kleiber, Anita Viszket (2003): Total Lexicalism and GASGrammars: A Direct Way to Semantics. In Gelbukh, A. (ed.): Proceedings of CICLing2003 (Mexico City). LNCS N2588. Springer-Verlag, Berlin Heidelberg New York. 37-48
8. Alberti, Gábor, Judit Kleiber (2004): The GeLexi MT Project. In J. Hutchins (ed.): Proceedings of EAMT 2004 Workshop, Valletta: Univ. of Malta, 1-10
9. Alberti Gábor, Kleiber Judit, Viszket Anita (2003): GeLexi Projekt: GEneratív LEXIkonon alapuló mondatelemzés. Csendes D., Alexin Z. (szerk.): MSZNY 2003, Szegedi Tudományegyetem, Egyetemi Nyomda, 79-85
10. Bódis Zoltán, Kleiber Judit, Szilágyi Éva, Viszket Anita (2003): Leíró nyelvtan - adatbázisból. Csendes D., Alexin Z. (szerk.): MSZNY 2003, Szegedi Tudományegyetem, Egyetemi Nyomda, 300-302
11. van Eijck, Jan, Hans Kamp (1997): Representing discourse in context. In van Benthem, J., ter Meulen, A. (eds.): Handbook of Logic and Language. Elsevier, Amsterdam, The MIT Press, Cambridge, Mass.
12. Karttunen, Lauri (1986): Radical Lexicalism. Report No. CSLI 86 68, Stanford
13. Mitkov, Ruslan ed. (2003): The Oxford Handbook of Computational Linguistics,. Oxford University Press.
14. Prószéky, Gábor, László Tihanyi, Gábor Ugray (2004): Moose: a robust high-performance parser and generator. In J. Hutchins (ed.): Proceedings of EAMT 2004 Workshop, Valletta: Univ. of Malta, 138-142

Hunglish: nyílt statisztikai magyar-angol gépi nyersfordító

Halácsy Péter*, Kornai András**, Németh László*, Rung András*, Szakadát István*, Trón Viktor***, Varga Dániel*

Kivonat A Budapesti Műszaki Egyetem Média Oktató és Kutató Központjának vezetésével 2004 júliusában indult Hunglish projekt[1] egy szabadon felhasználható, statisztikai gépi nyersfordítót, illetve fordítástámogató rendszert hoz létre, magyar nyelvű szövegek angolra való átültetéséhez. A gépi fordító tanításához egy kétnyelvű illesztett párhuzamos korpuszt hozunk létre. A projekt lezárása után nemcsak a kifejlesztett szoftvereket, hanem a korpuszt és az ez alapján épített/javított kétnyelvű magyar–angol szótárat is szabadon hozzáférhetővé tesszük bárki számára.

1. Bevezetés

A globális szolgáltatók szemszögéből a helyi nyelv használata elengedhetetlen termékeik és szolgáltatásaik új piacokra történő bevezetéséhez és elterjesztéséhez – különösen a termékleírások és az információ-szolgáltatások követelnek állandó fordítási munkát. A lokális piacok, a nemzeti kultúrák szemszögéből tekintve azonban más összefüggések válnak fontossá! Az információáramlás és az ebből fakadó gazdasági előnyök biztosítása érdekében elsősorban arra van szükség, hogy a helyben rendelkezésre álló információ globálisan elérhető legyen. A magyar viszonyokra vetítve tehát kulcsfontosságúnak tartjuk azt, hogy a magyar termékek, szolgáltatások és általában magyar nyelven elérhető információk minél hatékonyabban és minél szélesebb körben válhassanak ismertté. Ahhoz, hogy magyar nyelvű információ más nyelven is elérhető legyen, tömérdek fordítási munkára van szükség. Miután az angol nyelv mind a gazdasági életben, mind az információáramlásban központi szerepet kap, úgy gondoljuk, hogy a magyar nyelvből való gépi fordítás szempontjából az angol a kulcsfontosságú célnyelv. A projekt elsődleges célja így egy magyar-angol nyersfordító rendszer építése.

Nem tekintjük célunknak a magas szintű, netán irodalmi igényű gépi fordítást. Célunk olyan rendszer elkészítése, melynek kimenete egynyelvű

* Budapesti Műszaki Egyetem Média Oktató és Kutató Központ, {hp, nemeth, runga, szakadat, daniel}@mokk.bme.hu

** MetaCarta Inc., andras@kornai.com

*** International Graduate College, Saarland University and University of Edinburgh, v.tron@ed.ac.uk

[1] A projekt indulását az Informatikai és Hírközlési Minisztérium ITEM 2003 pályázatán elnyert összeg biztosítja.

információ-visszakereső (IV, angolul information retrieval) rendszerek bemeneteként szolgálhat. A többnyelvű IV rendszerek kutatásai, különösen az Amerikai Szabványügyi Hivatal (NIST) által évente megrendezett TREC konferencia „keresztnyelvi IV" (cross-language information retrieval) vizsgálatai világossá tették, hogy az IV rendszerek maguk sem képesek a finom árnyalatok megkülönböztetésére, és lényegében ugyanazt a teljesítményt nyújtják gyengébb minőségű (pl. beszédfelismerésből származó, 25-30%-ban hibás) szövegeken, mint a hibátlan nyelvtanú, választékosan megírt anyagokon. Ez annyit jelent, hogy nyersfordítás bizonyos használati helyzetekben ugyanolyan hasznos, mint egy igényes emberi fordítás.

A projekt végeredményeként egy működőképes nyersfordító szolgáltatás prototípusa fog elkészülni. A szoftvereket, vagyis a fordítóprogram kódját és a munka során kifejlesztett eszközkészletet, valamint a felépített adatbázisokat, a kétnyelvű illesztett korpuszt és a kétnyelvű szótárat szabadon hozzáférhetővé tesszük. A munka során kidolgozott módszereket és technológiát publikációk, illetve használati kézikönyvek formájában kiadjuk. A projekt eredményeit ezáltal bárki elérheti, felhasználhatja, illetve továbbfejlesztheti, vagy a technológiára építve önálló szolgáltatást indíthat.

Az eredményekhez való szabad hozzáférés a projekt egyik kulcsfontosságú eleme, amellyel számos célunk van. Egyrészt így látjuk biztosítva, hogy a támogatás megszűnésével a fejlesztések tovább folytatódhatnak, akár a jelen projekt résztvevőitől teljesen függetlenül is. Másrészt, minden olyan kutató- és fejlesztőcsoport munkáját támogatni kívánjuk, amely valamilyen módon a magyar nyelvtechnológiával foglalkozik. A projekt olyan alapvető fontosságú technológiai megoldásokat és adatforrásokat tesz hozzáférhetővé, melyek mind további alapkutatásokhoz, mind gyakorlati alkalmazások fejlesztéséhez elengedhetetlenek.

2. A projekt céljai

A gépi fordítás lényegében a számítógép megjelenésével egyidős vállalkozás; az első ilyen célú programot 1947-ben fejlesztették ki Weaver és munkatársai. A gépi fordítás nehézségeit összegző ALPAC jelentés[2] megállapításai sok tekintetben máig érvényesek, és emiatt nem meglepő, hogy a gépi fordítás alkalmazási köre meglehetősen korlátozott. Köztudomású, hogy a gépi fordító rendszerek kimenete kézi utószerkesztés nélkül emberi kommunikációra nem alkalmas, az automatikus fordítások gyakran kifejezetten komikus hatást keltenek. Éppen ezért jelen projekt célja sem az elsődlegesen emberi fogyasztásra szánt végleges fordítás, hanem csak a gépi vagy utószerkesztői felhasználásra szánt nyersfordítás.

Ehhez a főcélhoz vezető munkálataink során a projekt több olyan részeredményt is felmutat majd, amelyek önmagukban is jelentős nyelvtechnológiai hozzájárulásként tekinthetőek:

- magyar-angol szótár: szabad felhasználású, gyakorisági információkat is tartalmazó elektronikus magyar-angol szótár
- a statisztikai alapú szótárak előállításához, karbantartásához és javításához szükséges infrastruktúra

- párhuzamos korpusz: szabad felhasználású, mondatonként illesztett magyar-angol párhuzamos szövegkorpusz
- nyersfordító: szabad forrású rejtett Markov modell alapú nyersfordító technológia

A nyersfordítás legfontosabb eszköze a kétnyelvű szótár. Immár harminc éve vannak forgalomban olyan fordítástámogató rendszerek, melyek elsősorban a szavak szótári kikeresésének munkáját automatizálják. Projektünk *első célja egy jogtiszta, szabadon felhasználható magyar-angol szótár publikálása*, amelyet az egyéni felhasználók és a szoftverfejlesztő közösség szabadon bővíthet tovább. Ehhez komoly hozzájárulás Vonyó Attila közismert kétnyelvű gépi szótára. Amennyiben a magyarországi K+F-támogatási rendszer keretében további angol-magyar rendszerek is épülnek, és amennyiben az alkotók hajlandók ezek szóanyagát is nyílt forráskódúvá tenni (ideértjük nemcsak a kutatási, hanem a kereskedelmi célra való továbbfelhasználás korlátozás nélküli engedélyezését is), annyiban rendszerünk szótára ezekkel tovább bővíthető.

A szótári ekvivalencián alapuló (nyers)fordításnak ragozott szavak és szótári tételek problémáján kívül két alapproblémával kell megküzdenie. Az első probléma a célnyelv és tárgynyelv nyelvtani eltérései. Esetünkben ez különösen nagy problémaként jelentkezik az angol és a magyar nyelvi rendszer jelentős különbségei miatt. Amit az angol tipikusan szórendiséggel fejez ki (pl. az alany/állítmány/tárgy megkülönböztetést) azt a magyar ragokkal érzékelteti. Miután célunk elsősorban a gépi IV-t támogató nyersfordítás, a probléma nagyobb részét – elsősorban az angol szórend finomságainak algoritmizálását – mi figyelmen kívül hagyhatjuk, hiszen az információ-visszakereső rendszerek eleve a szöveg sorrendiségét elhanyagoló „szózsák" (angolul bag of words) modelleken alapulnak.

Egy másik probléma a szótári többértelműség. Például a magyar *nap* szó egyszerre jelenti az égitestet és az időegységet, amelyet az angol nyelv két külön szóval fejez ki (*sun*, illetve *day*). Miután egy magyar szónál átlagban három angol ekvivalenssel is lehet számolni, egy hétszavas magyar mondat lefordítása 3^7 (tehát több mint kétezer) variánst kínál. Erre a problémára megoldást nyújt a szövegkörnyezetben található információ, például abban a kifejezésben, hogy 'a nap és bolygói' a *nap* szó egyértelműen a *sun*, míg abban, hogy 'egy esős nap' egyértelműen a *day* fordítást kaphatja. Világos, hogy az ilyen környezettől függő valószínű fordítások megtalálásához szükséges, hogy a szövegkörnyezet által nyújtott információt pontosan meg tudjuk ragadni és azt elvszerűen integráljuk a potenciális ekvivalensek kiválasztásának folyamatában.

A nyelvi elemek egymás környezetében való megjelenésének statisztikai elméletét még a múlt század elején alkotta meg A. A. Markov. Ma ennek az elméletnek különféle változatai léteznek: a Markov-láncok (angolul Markov chains) és az ún. rejtett Markov modellek (HMM, angolul Hidden Markov Model) a nyelvtechnológia számos ágának alapvető eszközei, ezek közül külön kiemeljük a beszédfelismerést és a HMM alapú gépi fordítást[1]. A Markov modellezés nyelvtechnológiai használhatóságát a franciától a kínaiig már számos nyelvhez készült

alkalmazás bizonyítja. *A projekt második célja tehát a rejtett Markov modell technológiának alkalmazása a szótári többértelműség problémájának megoldására.*

A statisztikai módszer – bár kétségkívül eredményesebb, mint a hagyományos szabályrendszereken alapuló GF – azért nem csodaszer. Legfontosabb gyengesége abban áll, hogy a rendszer megépítése kifejezetten sok adatot igényel. A statisztikai alapú gépi fordítás alapvető adatforrása a párhuzamos korpusz. A párhuzamos korpusz olyan szövegminta, amely egy adott tartalmat két nyelven jelenít meg és a nyelvi egységek (például mondatok) sorrendileg illesztve vannak egymáshoz. *A projekt harmadik célja magyar-angol párhuzamos korpusz létrehozása.*

Párhuzamos kétnyelvű szövegkorpusz készítésének bevett módja szépirodalmi szövegek és igényes műfordításaik gyűjtése és illesztése. Ez a statisztikai alapú GF módszerhez szükséges adatmennyiségnek csupán töredékét (néhány száz megabyte-ra tehető anyagot) képes nyújtani. Ennél nagyobb baj, hogy az elérhető irodalmi jellegű források (pl. a Biblia vagy Orwell 1984 című regénye) a gyakorlati (nyers)fordításhoz nem megfelelőek. Mivel a gyakorlati gépi fordítás legfontosabb célszövegei üzleti, technológiai és jogi tartalmak, elengedhetetlen, hogy a szövegkorpusz ezeknek a területeknek a jellemző szakszókincsét minél nagyobb mennyiségben tartalmazza. A cél nem lehet Mikszáth angolra fordítása, hiszen ilyesmire vállalkozni automatizált módszerrel egyszerűen sarlatánság lenne. Praktikus lehet viszont, hogy a magyarul kiírt tenderek angol nyelven is elérhetőek legyenek, ami lehetővé tenné a beszállítók körének növekedését, és a magyar vevő potenciálisan több és jobb ajánlat közül választhatna. A korpusz előállításánál így elsősorban nem a szépirodalmi szövegekre, hanem a világhálón található többnyelvű szerverekre koncentrálnánk (l. [3]). Előzetes becsléseink szerint ettől egy nagyságrenddel nagyobb, és persze gyakorlati szempontból sokkal hasznosabb, párhuzamos korpusz várható.

Hivatkozások

1. Brown, Peter F. , Della Pietra, Stephen, Della Pietra, Vincent J., Mercer, Robert L.: The Mathematic of Statistical Machine Translation: Parameter Estimation. In Computational Linguistics 19 (1994) 263–311.
2. ALPAC 1966: Languages and machines: computers in translation and linguistics. A report by the Automatic Language Processing Advisory Committee, Division of Behavioral Sciences, National Academy of Sciences, National Research Council. Washington, D.C.: National Academy of Sciences, National Research Council. (Publication 1416.).
3. Resnik, Philip: Mining the Web for Bilingual Text. Proceedings of the International Conference of the Association of Computational Linguistics. Maryland. (1999)

A MetaMorpho projekt 2004-ben

Tihanyi László

MorphoLogic Kft.
1126 Budapest Orbánhegyi út 5.
tihanyi@morphologic.hu

Kivonat: A MetaMorpho fejlesztések idén, a projekt ötödik évében elérték a piacképes szintet. A programnak, mely a tervek szerint a kezdő, a haladó és hivatásos fordítói igényeket különböző termékekkel kívánja kiszolgálni, először a nyelvet nem vagy csak kezdőszinten beszélőknek ajánlott megértéstámogató változata, a MoBiCAT került a piacra. Ez a cikk folytatása az előző MSZNY konferencián elhangzott beszámolónak [1], és összefoglalja a 2004-es év legfontosabb fejlesztéseit.

1 Bevezetés

A idén június elején piacra került *MoBiCAT*, egy olyan internetes angol-magyar megértéstámogató fordítóprogram, amely a fordítást egy központi szerverrel végezteti. A MoBiMouse-hoz hasonló felülettel rendelkező programot Microsoft Word, Internet Explorer, Outlook és Adobe Acrobat alkalmazásokban használhatjuk. A fordítást az egérrel való rámutatással kezdeményezhetjük, és az eredeti mondat a fordításával együtt egy buborékban jelenik meg. A program ingyenesen letölthető és kipróbálható. Továbbiak a programról a www.mobicat.hu oldalról tudhatók meg.

Ugyancsak nyár elejére készült el a *morphoWAP*, a MetaMorpho mobiltelefonokról elérhető WAP-os kliense. A fordítás mobilszolgáltatóktól független, és jelenleg mindenki díjmentesen használhatja. Az eszköz egy-egy szó esetén szótárprogramként, mondatokra fordítóprogramként működik. A morphoWAP beépített angol helyesírásellenőrző programot is tartalmaz. A szolgáltatás a http://www.metamorpho.hu/wap címen érhető el wapos telefonokról.

Az idei fejlesztések harmadik tervezett alkalmazása az egész weboldalakat fordító *morphoWEB* fordítóprogram. Ez a program az angol oldalak magyar változatát az eredetivel azonos megjelenésben állítja elő az. Az oldalon lévő hivatkozásokat feloldja, tehát azokon tovább haladva ezeket is magyarul olvashatjuk.

A fokozatos minőségi javulás mögött számos nyelvészeti és szoftverfejlesztési újítás húzódik meg, ezek közül a fontosabbakat a következő fejezetekben mutatjuk be.

2 Nyelvészeti fejlesztések

Idén nyelvi fejlesztésünk eljutott arra a szintre, amikor a fő problémát már nem a szerkezetek leírása és felismerése, hanem a megfelelő minták érvényre juttatása kezdte okozni. A projektben kezdetektől fogva alapvető elv volt, hogy az elemzések számát a lehetséges legkisebb értéken tartjuk. Teljes lefedettség esetén minden eredményt, részleges elemzéskor pedig a részelemzések sorozatát jelenítjük meg. [2]

A többértelműség mindkét esetben problémát jelent, hiszen teljes elemzéskor a sok mondat lesz nehezen áttekinthető, részleges elemzéskor pedig minthogy önkényesen választjuk ki a többértelmű részelemzések egyikét, és fűzzük össze a teljes fordítást, a helyes elemzés gyakran meg sem jelenik.

A megfelelő minták érvényre juttatásának érdekében az eddig homogén lexikont és grammatikát szintekre tagoltuk. A szabályok leírásmódja és működési elve nem változott, de a megfelelő szintek olyan szűrőkként működtethetők, amelyek már csak az általuk megengedett eredményeket engedik tovább. Az új eljárással nemcsak a nyelvi minőség javulását, hanem a felesleges elemzések elkerülésével jelentős sebességbeli növekedést is el tudtunk érni. A fejlesztéséhez természetesen a szintaktikai elemző programot is megfelelően át kellett alakítani.

Egy másik jelentős fejlesztés az entitások, különösen a nagybetűs tulajdonnevek (néventitások) kezelésének megoldása volt.

További nyelvészeti fejlesztések voltak a koordináció kezelése, a globális tulajdonságok kezelése (cím, stb.), a címnyelvtan, az alárendelő mellékmondatok valamint a kötőmód helyes kezelése, és természetesen a folyamatos szótárbővítés.

3 Programfejlesztések

A programfejlesztés idei legfőbb eredménye, hogy létrejött egy interneten át egy egyszerre több felhasználót kiszolgálni képes, stabil szerver program. A szerver jogosultságkezelő védelmi modullal van ellátva, amely a fizetős használat mellett ingyenes kipróbálásra is lehetőséget ad.

Készült egy mintagyártó felhasználói program (RuleBuilder), amely képes arra, hogy forrásszövegekből olyan mintákat állítsunk elő, amelyet a rendszer a sajátjaival egyenértékű módon tud kezelni. Az eltárolt minták az SQL adatbázisban a nyelvi adatok mellett egy domainkezelő segítségével szakterületi és egyéb jelentésmegkülönböztető tulajdonságokkal is elláthatók. Az adatbázis közösen tárolja a felhasználói mintákat, de alapértelmezésben mindenki csak a saját mintáját látja. A rendszer felügyelőjének azonban lehetősége van, hogy bizonyos mintákat az egész közösséggel megosszon. Egyelőre házon belül használjuk lexikális bővítésekhez, de célunk, hogy mielőbb nyilvánossá tegyük, és a felhasználók számára lehetővé váljon a rendszer on-line bővítése.

Elkészült egy címazonosító modul [3], amire nagy volt szükség mert a MoBiCAT-et elsősorban internetes böngészéshez ajánljuk, és hírportálokon a felhasználó először címekkel találkozik. Az angol címek nyelvtana mint tudjuk, nagyon eltér a mondatszintaxisban leírtaktól.

Integráltuk a Brill-taggert. Statisztikai adatbázisait átvettük, de a szófaji egyértelműsítésre szolgáló eszköz kiértékelő függvényeit újraírtuk.

Kiemelendő még a jelentés-egyértelműsítő modul integrálása a programba [4]. A WEKA nevű Java alapú WSD program kiértékelő modulját C++ nyelven újraírtuk, és megkezdtük az egyértelműsítéshez szükséges minikorpuszok összeállítását. Ezekben a kérdéses szavak jelentése a WordNet rendszernek megfelelően meg van különböztetve, és a program a szó környezetében álló szavak gyakoriságából valószínűsíti az aktuális jelentést.

Minden modulunk (a MetaMorpho szerver, a MoBiCAT, a morphoWord, a morphoWeb és a RuleBuilder kliensek is) saját telepítőprogramot kapott, ami nemcsak a frissítéseket könnyíteti meg, hanem a verziószám megjelenésével pontosabb felhasználói visszajelzésekre ad lehetőséget.

4 Webfejlesztések

Elkészültek weboldalaink: www.metamorpho.hu, és http://www.mobicat.hu, amelyek tájékoztatást nyújtanak a projektről általában, és naprakész információval szolgálnak (letöltési oldal, hírek, GYIK). Ezekre az oldalakra érkeznek a felhasználói visszajelzések, és itt van lehetőség a program minőségének véleményezésére is. Elkészültek az internetes értékesítéshez szükséges webes felületek, és a MoBiCAT visszajelzések kezelésére szolgáló Internet oldal.

5 Tervek 2005-re

A jövő év nagy vállalkozása a magyarról angolra fordítás lesz. A 2005 elején induló NKFT által támogatott projekt a MTA Nyelvtudományi Intézetével és a Szegedi Tudományegyetem Informatikai Tanszékcsoportjával közös együttműködésben fog megvalósulni.

Jövő évi terveink között szerepel a hagyományos, szövegszerkesztőkben is használható fordítóprogram piacra bocsátása. Ugyancsak tovább folynak a MetaMorpho alapú fordítómemória fejlesztések, a szövegpárhuzamosítása [5] és a hasonlósági keresésre [6] irányuló kutatások.

Referenciák

1. Tihanyi László: A MetaMorpho projekt története. *I. MSzNy, Szeged* (2003)
2. Gröbler Tamás: Egyértelműsítés a MetaMorpho rendszerben *II. MSzNy, Szeged* (2004)
3. Pohl Gábor, Ugray Gábor: Angol címek felismerése *II. MSzNy, Szeged* (2004)
4. Miháltz: Gépi fordítórendszer támogatása jelentés-egyértelműsítő rendszerrel *II. MSzNy, Szeged* (2004)
5. Pohl Gábor: Iteratív bekezdés- és mondatszinkronizáció. *II. MSzNy, Szeged* (2004)
6. Hodász: Nyelvi hasonlóságon alapuló keresés fordítómemóriában *II. MSzNy, Szeged* (2004)

Egyértelműsítés és „mozaikfordítás" a MetaMorpho rendszerben

Gröbler Tamás

MorphoLogic kft.
1126 Budapest, Orbánhegyi út 5.
grobler@morphologic.hu

Kivonat: A gépi fordító rendszerekben több szinten is szükség van egyértelműsítésre. A MetaMorpho rendszerben különös jelentősége van az egyértelműsítésnek akkor, amikor a teljes mondat szintaktikai elemzése sikertelen, de számtalan – jellemzően egymást átlapoló – részelemzés keletkezik, amelyekből össze kell állítani a teljes mondat lehető legjobb fordítását. Ezt az eljárást nevezzük *mozaikfordításnak*. A cikk összefoglalja azokat a szintaktikai elemzést kiegészítő heurisztikákat, amelyeket a legjobb mozaikfordítás előállításának érdekében alkalmazunk.

1. Bevezetés

A nyelvi elemzés szempontjából általában a többértelműség két típusát szokás megkülönböztetni: a lexikális és a strukturális többértelműséget. Ezek kezelésének hagyományos, „tankönyvi" receptje szerint a lexikális többértelműség feloldása a szintaktikai elemzés előtt, a strukturális egyértelműsítés pedig az után következik.

A MetaMorpho angol-magyar gépi fordítórendszerben [3] ettől eltérő megközelítést alkalmazunk. Az egyes mondatok fordításakor abból indulunk ki, hogy az általunk kézben tartott és bármikor fejleszthető nyelvtanra épülő szintaktikai elemzés sikere, vagyis a teljes mondat szerkezetének feltárása előnyt élvez minden más egyértelműsítéssel szemben. Ritkán előfordul, hogy egy mondatot többféleképpen is sikeresen elemez meg a nyelvtan, de ilyenkor az összes megoldást jónak tartjuk. Ezért a szintaktikai elemzés előtt nem végzünk egyértelműsítést, és ha a teljes mondat elemzése sikeres, akkor utána sem.

Vizsgálatunk tárgya tehát az az eset, amikor a teljes elemzés nem áll elő, hanem a mondat egyes részeire épülő elemzések sokaságából kell összeválogatni azokat, amelyek

a.) a legjobb fordítást eredményezik,
b.) nem átfedők,
c.) lehetőleg lefedik a teljes mondatot.

Ezt az eljárást nevezzük *mozaikfordításnak*, amely során nagymértékben felhasználjuk az egyes egyértelműsítési eljárások eredményeit.

2. Egyértelműsítés a MetaMorpho rendszerben

A MetaMorpho fordítórendszer jelenleg a következő egyértelműsítési eljárásokat tartalmazza:

- szófaji egyértelműsítés
- jelentés szerinti egyértelműsítés
- a lexikális egységek határainak egyértelműsítése
- szintaktikai egyértelműsítés

Míg az utóbbi két eljárás alapvetően a nyelvtan részeként működik, a szófaji és a jelentés szerinti egyértelműsítést külön modulok (POS tagger, ill. WSD modul [2]) végzik, amelyek a hozzájuk kapcsolódó *szűrőkön* keresztül épülnek be az elemzés folyamatába [4].

A nyelvtan számos olyan mechanizmust tartalmaz, amelyek a szintaktikai szerkezetet egyértelműsítik, és ezáltal biztosítják, hogy a tipikusan több ezer létrejött elemzési tényből általában csak néhány tucat gyökérelem marad, amelyek közül válogatni kell. Ilyen mechanizmus például a tények egymás közötti „ölési mechanizmusa" és a gyökérszimbólumok kitüntetése a tények között [3].

A nyelvtan többszintűsége lehetővé teszi a szintenkénti egyértelműsítést is, például a többszavas lexikális egységek (szóösszetételek, számok, dátumok, földrajzi, intézmény- és személynevek) felismerését, és ezáltal a lexikális elemek határainak egyértelműsítését. (A korábbi változatban ezt a feladatot is egy szűrő végezte.)

A jelentés-egyértelműsítő szűrőt [2] mutatja be. Ez annyiban különbözik a többitől, hogy az elemzések számát nem csökkenti, hanem az egyes részelemzésekből generálható többes fordítás lehetőségét szünteti meg.

A mozaikfordítás előállításához szükséges további egyértelműsítéseket az utolsó fejezetben bemutatott szűrők végzik.

3. A mozaikfordítás előállítása

A mozaikfordítást a gyakorlatban úgy valósítjuk meg, hogy a forrásnyelvi szintaktikai elemzés eredményét *szűrőkön* engedjük keresztül [4]. Mivel bizonyos esetekben szükségünk lehet az összes részelemzésre, a legjobb fordítás érdekében először egy *rendező szűrőt* alkalmazunk, amely az egyértelműsítő modulok segítségével az alábbiak szerint állítja sorba a részelemzéseket.

A rendező szűrő bármely két részelemzésről képes eldönteni, hogy melyiket részesíti előnyben. Az összehasonlítandó két elemzést az előre megadott rendezési szempontok szerint, azok szigorú sorrendjében hasonlítja össze. A szempontok sorrendjében később következő szempontokat csak akkor vesszük figyelembe, ha az előttük álló egyik szempont szerint sem eldönthető, hogy melyik részelemzés a jobb. Az egyes rendező szűrők működését a következő fejezet mutatja be.

A mozaikfordításokat egy külön szűrő állítja össze a már rendezett részelemzésekből a b.) és c.) követelményeknek megfelelően. Az egyes lefedéseket pontozzuk aszerint, hogy az előzetes rendezés szerint mennyire jó mozaikokból állnak. Egy lefedés pontszámát úgy számítjuk ki, hogy összeadjuk a benne szereplő részelemzéseknek (mozaikoknak) a rendezés során kialakult sorszámát. Ezáltal a kevés és „jó" mozaik-

ból álló fordítás pontszáma lesz a legkisebb. Mivel a lehetséges lefedések száma exponenciálisan függ a részelemzések számától, ezt a válogatást legalább részben „mohó" módon végezzük, vagyis a legjobb elemzésektől indulva csak előre rögzített számú lefedés pontszámát számítjuk ki. Ha csak a legjobb mozaikfordításra vagyunk kíváncsiak, akkor a legalacsonyabb pontszámú lefedést választjuk.

4. Egyértelműsítő szűrők

Az alábbi példákban a MetaMorpho rendszerben jelenleg megvalósított szűrőket mutatjuk be. A szűrők működését egy-egy példával is illusztráljuk. A példákban olyan mondatokat kell vizsgálnunk, amelyek teljes elemzése valami miatt meghiúsul. Mivel ezek általában hosszúak, az áttekinthetőség kedvéért csak azokat a kifejezéseket tüntetjük fel, amelyeken a szűrő hatása a legjobban megfigyelhető. Az egyértelműsített elemzés bemutatására a magyar fordítás látszott a legalkalmasabbnak. A fordításban szögletes zárójel jelöli az egyes mozaikok határát.

4.1. Szófaj-egyértelműsítő szűrő

Ehhez a szűrőhöz külön modulként implementáltuk Brill [1] szófaji egyértelműsítőjét (POS tagger). A szűrő az egyértelműsítés által kijelölt szófajokból épülő részelemzéseket részesíti előnyben. Ha két részelemzés azonos tartományt fed le, akkor az „győz", amelyik kevésbé tér el az egyértelműsített szófaj-sorozattól. A nem azonos tartományt lefedő részelemzéseknél azonban óvatosan csak az olyan eltérést vesszük figyelembe, ahol az egyik szófaj ige (de nem gerund vagy perfect alak), a másik pedig névszó (főnév, melléknév vagy határozószó).

	szűrő nélkül	szűrővel
angol	*the dog barks*	
magyar	*[a kutyakérgek]*	*[a kutya ugat]*

4.2. Tagmondat-kiválasztó szűrő

Sokszor azt szeretnénk, hogy ha a teljes mondat nem is áll össze, a felismert tagmondatok mindenképpen szerepeljenek a mozaikfordításban. Ezek helyett azonban sokszor más részelemzések is előjönnek, és ilyenkor a szófaji egyértelműsítés is tévedhet. Ezért az önállóan tagmondatot alkotó részelemzéseket előnyben részesítjük.

	szűrő nélkül	szűrővel
angol	*the plane lands*	
magyar	*[a repülőgépföldek]*	*[a repülőgép leszáll]*

4.3. Töredék-szűrő

Mivel semmiképpen sem szeretnénk, hogy bármely szó is kimaradjon az elemzésből, minden szóhoz tartozik egy részelemzés, amely végső esetben „beugrik" az esetleg kimaradó pozícióba. Az ilyen részelemzéseket töredékeknek hívjuk. A töredék-szűrő gondoskodik róla, hogy a töredékek a lista aljára kerüljenek, és csak akkor kerüljenek be a mozaikfordításba, ha nincs más elemzés. A példában a római számokból képzett töredéket bírálja felül a személyes névmásból létrejött főnévi csoport.

	szűrő nélkül	szűrővel
angol	*I*	
magyar	*[I]*	*[én]*

4.4. Pozicionális szűrő

Ha a részelemzések sorrendjét a fenti szűrők egyike sem tudja meghatározni, akkor azok egymáshoz viszonyított helyzete alapján állapítjuk meg a sorrendet. Ebben az esetben a leghosszabb elemzéseket részesítjük előnyben, ezek közül pedig a balrább kezdődőket választjuk.

	szűrő nélkül	szűrővel
angol	*the train stations in poor countries*	
magyar	*[a vonat] [állomásoztat] [] [szegény országok]*	*[a vonatállomások szegény országokban]*

5. Összefoglalás

A bemutatott egyértelműsítési eljárások segítségével elértük, hogy a mozaikfordítás minősége jelentősen javult. A továbbiakban a legtöbb lehetőséget a szófaji egyértelműsítésben látjuk. Ennek finomításától és más POS tagger algoritmusok kipróbálásától további javulást remélünk.

Irodalom

1. Brill, E. (1992): 'A simple rule-based part of speech tagger'. *Proceedings of the Third Conference on Applied Natural Language Processing*, ACL, Trento, Italy.
2. Miháltz M. (2004): 'Angol-magyar gépi fordítórendszer támogatása jelentés-egyértelműsítő rendszerrel'. *ebben a kötetben*
3. Prószéky G., Tihanyi L. (2002): 'MetaMorpho: A Pattern-Based Machine Translation Project'. *Translating and the Computer 24*, ASLIB, London.
4. Prószéky G., Tihanyi L., Ugray G. (2004): 'Moose: A Robust High-Performance Parser and Generator'. *9th EAMT Workshop*, Malta.

Angol-magyar gépi fordítórendszer támogatása jelentés-egyértelműsítő modullal

Miháltz Márton

MorphoLogic Kft.
1126 Budapest, Orbánhegyi út 5.
mihaltz@morphologic.hu

Kivonat. A számítógépes jelentés-egyértelműsítés során egy adott nyelven többértelmű lexikai elemekről kell eldönteni, hogy adott előfordulásuk kontextusában az ismert jelentéseik közül melyekkel szerepelnek. Ennek a feladatnak speciális esete, amikor a megkülönböztetendő jelentéseket nem egy egynyelvű szótár meghatározásai, hanem egy másik nyelven lehetséges különböző fordításaik alapján határozzuk meg. A cikkben bemutatott, Naiv Bayes osztályozó algoritmussal működő supervised egyértelműsítő rendszer egy angol-magyar fordítóprogram támogatásához készült. A jelenleg prototípus implementációban működő, 38 különböző többértelmű angol főnevet kezelő rendszer 84%-os átlagos pontossággal egyértelműsít.

1 Bevezetés

A számítógépes jelentés-egyértelműsítésnek (Word Sense Disambiguation, WSD) nevezett feladatban a számítógép feladata annak eldöntése, hogy egy adott nyelv lexikálisan többértelmű tételei egy adott szövegben, adott kontextusban az előzetesen meghatározott lehetséges jelentéseik közül melyekkel szerepelnek. Ebben a jelenleg igen aktívan kutatott problémában az egyik fő nehézséget a megkülönböztetendő jelentések megfelelő leírása jelenti. A többértelműség különböző formái különböző fokú nehézség elé állítják egy adott nyelv beszélőit: míg homonim alakok (pl. *körte*) megkülönböztetésekor az adott nyelvet anyanyelvként beszélő két ember közel minden esetben egyetért, addig a poliszémia termékeny, finom jelentésárnyalatbeli különbségeket produkáló eseteiben az egyetértés még anyanyelvi beszélők között is akár 85%-os vagy alacsonyabb is lehet ([4]). Számítógépes rendszereknél sem magasabb ez az érték: a 2001-es Senseval-2 nyilvános WSD versenyen főneveknél az angol lexikális minta feladatban legjobban teljesítő rendszer pontossága a jelentések durva felbontású, kevésbé megkülönböztető leírásával 76,6%, míg az árnyaltabb, több jelentést megkülönböztető, finomabb felbontású jelentés-leírással 69,5% volt ([5]).

Természetesen a jelentések leírásának módját egy adott WSD rendszer megvalósításában főképpen a rendszer konkrét célja határozza meg. Az alábbiakban bemutatott angol nyelvű jelentés-egyértelműsítő rendszerben a különböző, általában egynyelvű szótárak értelmezései alapján meghatározott lehetséges jelentéseket leképeztük azok különböző magyar fordításaira. Mivel az esetek nagy részében több különböző angol

jelentés is ugyanazzal a magyar fordítással fejezhető ki, az azonos fordítású jelentések összefogásával sikerült elérni egy durvább felbontású jelentés-leírást, ahol kevesebb, egymástól jobban elkülönülő alternatíva megkülönböztetésére van szükség.

A bemutatandó rendszer a MorphoLogic Kft. által fejlesztett *MetaMorpho* angol-magyar megértés-támogató fordítóprogram ([8]) támogatását szolgálja. A fordítórendszerben a többértelmű szavak különböző magyar fordítású jelentéseinek megkülönböztetésére van szükség, amikor a program nem képes csupán szintaktikai információk alapján egyértelműsíteni. Az alábbi két példában az előfordulási kontextus szemantikai tartalmának feldolgozására van szükség ahhoz, hogy a többértelmű angol *party* főnévhez a két esetben a megfelelő magyar fordításokat rendelhessük:

1. The **party** that won the elections four years ago did not make it into Parliament this time.
2. The **party** yesterday celebrated her birthday at one of the finest restaurants in town.

2 A jelentés-egyértelműsítő rendszer

2.1 Az osztályozó algoritmus

A bemutatandó jelentés- (vagy fordítás-) egyértelműsítő rendszer felügyelt (supervised), statisztikai osztályozó algoritmussal működik, így a betanításához szükség van kézi munkával, emberi erővel annotált tanítópéldákra. A statisztikai elven működő algoritmus mellett főképpen hatékonysági megfontolások miatt döntöttünk, hiszen a memória-alapú tanuló-rendszerekben, pl. a „lusta" KNN algoritmusban minden tanítópéldát állandóan „észben kell tartania" a rendszernek. Emellett pedig az ilyen rendszerben a különböző jegyek megfelelő súlyozása elengedhetetlen a helyes működéshez ([5]).

A rendszer az elterjedt, egyszerű *Naiv Bayes* osztályozó algoritmussal működik, mely a Bayes-tétel alapján, a különböző jegyek értékeinek egyes jelentésekre, mint osztályokra vett együttes feltételes valószínűségei alapján választja ki azt a jelentést, ami az adott kontextusban a legvalószínűbb. Az elnevezésben a „naiv" szó arra utal, hogy az algoritmus feltételezi, hogy az egyes kontextuális jegyek független valószínűségi változók. Ez természetesen természetes nyelvi szövegekben általában nem teljesül, azonban így is meglepően jó eredmények érhetők el ezzel a módszerrel ([3], [4]).

Az egyértelműsítő rendszer számára a tanuló algoritmuson felül alapvető fontosságú a többértelmű szavak előfordulási kontextusait reprezentáló jegyek helyes megválasztása is. Ezeknek a jegyeknek a tanítópéldákon megfigyelt értékeivel megy végbe a tanulás. Az ismeretlen jelentésű szó kontextusát szintén ilyen jegyekkel reprezentáljuk, illetve az online működés során ilyen jegyekből alkotott vektort osztályoz a Naiv Bayes algoritmus.

A bemutatott rendszer jelenleg a [3] és [5] munkákban bemutatott kontextuális jegyek egy részét használja, melyek két csoportba oszthatók. A csak az egyértelműsítendő szó mondatából kikerülő *lokális jegyek* főként a kontextus szintak-

tikai tulajdonságait, gyakori kollokációkat, módosítókat stb. reprezentálnak. A felhasznált lokális jegyek a következők:

- a többértelmű szó felszíni alakja (ahogy a szövegben előfordul)
- funkcionális szavak a többértelmű szó körül 2+2-es méretű ablakban
- tartalmas szavak tövei a többértelmű szó körül 3+3-as méretű ablakban

A jegyek másik csoportjába a teljes rendelkezésre álló kontextus topikjára, az aktuális szójelentés szemantikai tartományára jellemző *globális jegyek* tartoznak. Ezt az információt a tágabb kontextusban található, az adott tartalmas szóval leggyakrabban előforduló szavak előfordulási arányából alkotott vektorral reprezentáljuk (a sz0vegbeli lineáris sorrend és az egyértelműsítendő szótól való relatív távolság itt nem számít). A rendszerben paraméterezni lehet, hogy (a tanítópéldák alapján) az első hány leggyakoribb tartalmas szót, illetve hány, a többértelmű szó mondatát megelőző mondatot vegyen a kontextusból figyelembe.

Noha a rendszer jelenleg az összes ismert főnév esetében ugyanazokat a jegyeket használja, a rendszer implementációja lehetőséget biztosít arra is, hogy a későbbiekben a különböző szavakhoz automatikus jegy-kiválasztással ([5]) külön-külön megállapíthassuk a legoptimálisabb jegyhalmazokat.

2.2 A rendszer működése a fordítóprogramban

A Naiv Bayes osztályozóval működő egyértelműsítő rendszert a prototípus szinten a WEKA Java-s programrendszerrel ([10]) teszteltük, jelenleg folyik a saját fejlesztésű (C++ nyelven írt) motor tesztelése. Az osztályozó számára a lokális és globális jegyeket diszkrét, illetve numerikus értékkészletű attribútumokra képeztük le. A diszkrét attribútumok értékkészlet-halmaza a lokális jegyek tanítópéldákon megfigyelt értékekből áll, a numerikus attribútumok pedig egyenként jelölik, hogy a globális kontextus hányat tartalmaz a fontosnak ítélt tartalmas szavak közül. A numerikus attribútumok értékeit kernel sűrűség közelítéssel (kernel density estimator) modelleztük, amely nem feltételez normál eloszlást, és szignifikánsan javít a Bayes osztályozó pontosságán ([2]).

A rendszer számára kézzel annotált tanítókorpuszokhoz a Senseval ([1], www.senseval.org), és az Open Mind Word Expert ([6]) projektek jóvoltából jutottunk. A korpuszokat elsőként közös XML formátumba konvertáltuk, majd a nyelvi előfeldolgozási lépések következtek: bekezdésekre, mondatokra és szavakra bontás, morfológiai elemzés (HuMor angol morfológiai elemző), szófaji egyértelműsítés (MorphoLogic saját transzformációs szabályalapú POS-tagger), tövesítés. A tanítópéldák közül eltávolítottuk azokat, amelyek lexikálisan felismerhető, több szóból álló összetételeket tartalmaztak, mivel ezeket a fordítóprogram külön minták alapján, a WSD modul támogatása nélkül képes lefordítani. A feldolgozott korpuszokból kinyert kontextuális jegy-vektorokkal tanítottunk be Naiv Bayes osztályozókat, jelenleg 38 különböző többértelmű angol főnév egyértelműsítéséhez.

A MetaMorpho rendszerben az egyértelműsítő modul feladata az előfeldolgozott mondatokban (illetve bekezdésekben) az ismert többértelmű szavakon egy nyelvtani jegy értékének specifikálása, mely azok aktuális jelentését határozza meg. Ezen a ponton a jelentés-azonosító jegyek még az eredeti, többnyire a Princeton WordNet-ből ([7]) származó finom felbontású azonosítókat kapják értékül. A fordítóprogramban ezután történik a szintaktikai elemzés, ahol a szintaktikai szabályok már támaszkod-

hatnak a jelentés-azonosító jegyek értékeire is. Az elemzési fák felépülése után, az elemzéssel párhuzamosan kiválasztott generáló szabályok alkalmazásával létrejön a mondatok magyar fordítása. A magyar többértelmű főnevek generáló szabályaiban vannak az angol jelentések és magyar fordítások közötti leképezések elágazásszerűen rögzítve. Generáláskor a rendszer a jelentés-azonosító jegy WSD modul által beállított értéke alapján választja ki a megfelelő fordításokat. Ennek a megoldásnak két előnye van. Egyrészt, a magyar fordítások nincsenek „beledrótozva" az egyértelműsítő motorba, a működő rendszer nyelvtanában könnyen módosíthatjuk-javíthatjuk az angol jelentésekről való leképezéseket. Másrészt a megfelelő generáló szabályok megírásával nyitva áll az út az angolról más nyelvre fordító, jelentés-egyértelműsítő rendszerrel támogatott fordítóprogram létrehozása előtt is.

Az angol jelentésekről magyar fordításokra való leképezésben természetesen felmerülnek problémák is. Bizonyos többértelmű angol szavak egyes angol jelentései nem adhatók vissza egyetlen magyar fordítással, ilyenkor az eredeti jelentések tanítópéldáinak szétválasztására, magyar fordításokkal való újracímkézésére van szükség. A megvizsgált 38 többértelmű angol főnév között 4 olyat találtunk, ahol egyes jelentések esetén a fentiek szükségesnek bizonyultak volna (eddig azonban idő hiányában ezt nem tudtuk elvégezni, ezeket a jelentéseket kihagytuk a jelenlegi rendszerből).

Előfordult olyan eset is, amikor azért kellett kihagyni rendelkezésre álló szavakat, mert az összes angol jelentésük visszaadható volt egyetlen magyar fordítással (2 ilyen eset).

3 A rendszer értékelése

A jelentés-egyértelműsítő modul pontosságának megállapításához megvizsgáltuk a WSD-ben szokásos pontosság (precision) értékeket a jelenleg rendelkezésre álló 38 főnév tanítókorpuszán 10-szeres keresztellenőrzéssel (10-fold stratified cross validation). A pontosság a helyesen klasszifikált példák százalékos arányát jelenti az összes klasszifikált példához képest. A 10-szeres keresztellenőrzés során az egyes korpuszokat a jelentések eloszlási arányának megtartásával 10 egyenlő részre osztottuk, majd az egyes kis részeken végighaladva, a maradék részeken pedig tanítva teszteltük a rendszer pontosságát, és átlagoltuk az eredményt. Az alapszint minden esetben az adott szó leggyakoribb jelentésének relatív gyakorisága.

A teszteket elvégeztük mind az eredeti angol jelentésazonosítókkal, mind a magyar fordításokra leképzett és összevont azonosítókkal is. Az angol jelentés-azonosítók fölött történt egyértelműsítés eredményeit hely hiányában tételesen nem tudjuk közölni. Ennek átlaga a 38 főnév felett 76,39%-os pontosság 10-szeres keresztellenőrzéssel (64,15% átlagos alapszint). A magyar fordításokra történő egyértelműsítés eredményei az 1. Táblázatban láthatók.

Az egyértelműsítő algoritmus hibája a teljes tanítókorpuszon ellenőrizve angol jelentéseknél 1,90%, magyar fordításoknál 1,48%.

Az 1. Táblázatból leolvasható, hogy a megvizsgált 38 főnév esetében az egyértelműsítő pontossága általában meghaladta az alapszint értékét. Ez alól kivétel 5 eset, amikor egyenlő volt vele, illetve további 5 eset, amikor alacsonyabb volt annál (ekkor azonban csak átlagosan 1,77%-kal maradtak el az értékek az alapszinttől).

Az angol jelentések számának csökkentése a magyar fordításokra való leképezéssel átlagosan 7,86%-ot javított az egyértelműsítés pontosságán. 14 esetben nem változott így a pontosság, és csak egyetlen esetben csökkent (ebben az esetben mind az eredeti, mind az összevont jelentésazonosítók használatával a pontosság elmaradt az alapszinttől).

A tanítópéldák teljes mennyisége és a legritkább jelentéshez tartozó példák száma úgy tűnik együttesen befolyásolják a rendszer pontosságát. Az alapszint értékétől kevesebb, azzal egyenlő, vagy azt csak minimálisan meghaladó pontosságú esetekben az összes példák száma kb. 200-nál kevesebb, a legkevésbé reprezentált jelentéshez pedig kb. 20-nál kevesebb példa tartozott. Ennél magasabb együttes értékek jó eredményeket adtak.

4 További munka

A 38 főnév átlagos egyértelműsítési pontossága, ha a finom felbontású angol jelentésazonosítókat tekintjük, mindössze 0,21%-kal tér el a 2001-es SensEval-győztes rendszer pontosságától (ld. 1. rész). Ez a pontosság már a jelenlegi, hosszú jövőbeni fejlesztés előtt álló WSD rendszer esetében is alkalmas arra, hogy használható legyen egy olyan interaktív, emberi intelligenciára is építő rendszerben, mint a MetaMorpho megértés-támogató fordítóprogram.

A közeli jövőben szeretnénk a rendszerbe beledolgozni a WSD közösség számára rendelkezésre álló, szabadon hozzáférhető, illetve megvásárolható angol tanítókorpuszok anyagát is. Az egyértelműsítő szótárának további bővítéséhez szükség lehet egyéb szemantikailag annotált anyagok (pl. SemCor, Extended WordNet, MEANING projekt stb.) feldolgozására is. Ontológiák, szótárak felhasználásával, bootstrapping módszerek segítségével állíthatunk elő további tanítópéldákat ([3]), valamint hasznosak lehetnek ebből a szempontból a szinkronizált kétnyelvű (párhuzamos) korpuszok is.

Szintén a közeli jövőben szeretnénk a rendelkezésre álló korpuszok többértelmű igei és melléknévi anyagát is a rendszernek megtanítani. Az eltérő szófajok esetében valószínűleg különböző kontextuális jegyek bizonyulhatnak megfelelőnek, szükség lesz a 2.1 részben már említettekhez hasonló automatikus jegy- és paraméteroptimalizáció alkalmazására külön-külön az egyes többértelmű tételeknél.

A MetaMorpho rendszer szintaktikai elemzőjének fejlődésével szeretnénk továbbá a jelentés-egyértelműsítő modul számára a tanuláshoz és az egyértelműsítéshez magasabb nyelvi szintű információkat is rendelkezésre bocsátani. A tulajdonnév- és főnévi csoport-felismerés eredményeiből, illetve más morfoszintaktikai és szintaktikai információkból származó jegyek használatától a jelentés-egyértelműsítés pontosságának további javulását várjuk.

Bibliográfia

1. Edmonds, P. Kilgariff, A.: Introduction To The Special Issue On Evaluating Word Sense Disambiguation Systems. Journal of Natural Language Engineering 8 (4) (2002), 279-291

2. John, H. G., Langley, P.: Estimating Continuous Distributions in Bayesian Classifiers, in Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. Morgan Kaufmann, San Mateo (1995)
3. Leacock, C., Miller, G. A., Chodorow, M.: Using Corpus Statistics and WordNet Relations for Sense Identification. Computational Linguistics, Special Issue on Word Sense Disambiguation. (1998)
4. Manning, C. D., Schütze, H: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA (1999)
5. Mihalcea, Rada Word sense disambiguation with pattern learning and automatic feature selection. Journal of Natural Language Engineering (special issue on evaluating word sense disambiguation systems, 8 (4) 279-291 (2002)
6. Mihalcea, R., Chklovski, T.: Building a Sense Tagged Corpus with Open Mind Word Expert. Proceedings of the ACL-02 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions (2002)
7.. Miller, G. A., Beckwith R., Fellbaum, C., Gross D., Miller, K. J.: Introduction to WordNet: an on-line lexical database. International Journal of Lexicography 3(4) (1990) 235 – 244
8. Prószéky, G., Tihanyi, L.: MetaMorpho: A Pattern-based Machine Translation Project. Proceedings of the 24th 'Translating and the Computer' Conference. London, UK, 19–24 (2002)
9. Vancsa, L.: A „BLEU" automatikus kiértékelési eljárás alkalmazása angol-magyar fordítóprogram gyakori, folyamatos minősítésére. Magyar Számítógépes Nyelvészet Konfeencia, Szeged (2003)
10. Witten, I. H., Frank, E.: Data Mining: Practical machine learning tools with Java implementations. Morgan Kaufmann, San Francisco (2000)

1. Táblázat. A jelentés-egyértelműsítő validációjának eredményei a különböző többértelmű tételeken (átlagos pontosság 10-szeres keresztellenőrzés után)

Főnév	Jelentések száma		Tanítópéldák sz.		Alapszint	Pontosság
	Angol	Magyar	Összes	Legritkább magy. jelentéshez		
arm	5	4	787	16	56,67%	93,27%
art	4	2	108	3	97,22%	97,22%
authority	3	3	257	18	54,09%	68,09%
bank	4	2	398	7	98,24%	98,74%
bar	7	4	337	7	54,01%	60,53%
bum	5	2	118	20	83,05%	80,51%
chair	8	3	191	11	87,96%	87,43%
chance	6	4	615	21	65,37%	77,40%
chapter	3	2	137	45	67,15%	85,40%
child	7	2	180	66	63,33%	68,89%
church	3	2	183	76	58,47%	75,96%
circuit	6	4	184	25	43,48%	76,63%
day	2	2	192	67	65,10%	76,04%
degree	4	2	485	124	74,43%	96,29%
dyke	4	2	86	13	84,88%	87,21%
facility	3	2	37	2	94,59%	94,59%
fatigue	4	2	104	11	89,42%	93,27%
feeling	3	2	149	11	92,62%	90,60%
grip	5	2	218	17	92,20%	93,12%
hearth	3	2	96	17	82,29%	82,29%
holiday	4	2	83	3	96,39%	96,39%
image	7	2	512	219	57,23%	86,52%
lady	4	2	134	11	91,79%	92,54%
letter	3	2	927	140	84,90%	92,23%
line	5	4	4157	374	53,43%	84,94%
mouth	2	2	169	9	94,67%	93,49%
operator	2	2	119	31	73,95%	78,15%
party	2	3	623	108	42,05%	88,28%
performance	2	2	353	131	62,89%	88,95%
plane	4	3	474	2	96,41%	97,05%

post	3	3	141	18	63,12%	80,14%
process	2	2	302	70	76,82%	76,82%
report	3	3	335	42	67,76%	81,79%
restraint	6	4	89	2	44,94%	74,16%
sense	4	3	136	16	50,74%	55,88%
spade	5	3	89	4	71,91%	85,39%
stress	3	2	115	14	87,83%	85,22%
term	5	3	125	15	70,40%	80%
Átlag:					73,47%	84,25%

MemoQ – új megközelítés a fordítástámogatásban

Lengyel István, Kis Balázs, Ugray Gábor

Kilgray Kft.
{istvan.lengyel;balazs.kis;gabor.ugray}@kilgray.com

A fordítók körében a fordítástámogatási eszközök használatáról végzett felmérések (Drugan 2004, Fulford-Granell-Zafra 2004, Somers 2003) kimutatták, hogy csak kevés fordító alkalmaz teljes egészében számítógéppel támogatott munkafolyamatot. Ha azonban a fordítókat olyan közösségnek tekintjük, amelynek tagjai hasonló módon és hasonló szabályok, feltételezések alapján dolgoznak, és a fordítási folyamatot ennek alapján közelítjük meg, a termelékenységet tovább növelhetjük. A gyakorlatba ez holisztikus szemlélettel megvalósított, integrált fordítástámogató szoftver segítségével ültethető át. A MemoQ projekt az előadásban ismertetett megközelítés első implementációja.

1. Bevezetés

A legtöbb tudományos cikk a fordítást önállóan végzett tevékenységnek tekinti, és mint ilyent, a külvilágtól elszigetelve vizsgálja. A valóságban azonban a fordítók a piac törvényei szerint dolgoznak, ahol a hatékonyság még a minőségnél is fontosabb. A hatékonyság a hasznosság és a költségráfordítás közötti kapcsolatot határozza meg. A hasznosság a fordítás felhasználásából származik, így a legjobb fordítás – a piaci viszonyok alapján – az a fordítás, amely teljesen megfelel egy előre meghatározott, kimondott vagy ki nem mondott célnak.

A számítógéppel támogatott fordítás célja a hatékonyság növelése. Mindazonáltal a legtöbb ilyen rendszert lépcsőzetesen építették fel, és a fejlesztés első szakaszában mindegyik csupán az egyéni, íróasztala mögött egyedül dolgozó fordító igényeit elégítette ki. Az Internet később elsődleges kommunikációs csatornává és a leggyorsabb kutatási forrássá vált a fordítók számára, és mindez jelentősen csökkentette a piaci reagálási időt.

Az Internet azonban nem csupán a fordítási munkát alakította át, hanem további fordítási igényeket is támasztott. A fordítók kiszakadtak kreatív magányukból, és olyan helyzetbe kerültek, ahol minden és mindenki kapcsolatban van egymással, a határidők pedig sokkal szorosabbak, mint korábban.

A számítógépes fordítástámogató eszközök *(Computer-assisted Translation,* CAT) fejlesztői természetesen reagáltak az internet kihívására, de eszközeik – mivel nem írták át teljesen őket – nem tükrözik a paradigmaváltást. A fordítómemóriák továbbra is egygépes alkalmazások, minden felhasználónak egyforma lehetőségek állnak rendelkezésére.

A programok nem veszik figyelembe, hogy a fordítási projekten nemcsak fordítók dolgoznak, hanem projektmenedzserek, nyelvi vezetők, lektorok, kiadványszerkesztők

is. Minden felhasználónak mélységében kell ismernie a fordítómemóriákat, ha hatékonyan akar dolgozni, és a fordítómemóriák *(translation memory,* TM) elérhetőségében sem állt be nagy változás.

Amellett érvelünk, hogy hálózati környezetben lehetséges a munkamegosztás előnyeinek kihasználása, egyszerűsíthető a felhasználói felület, megakadályozható az adatvesztés – azáltal, hogy kevésbé kötjük a fordítómemóriát az operációs rendszer fájlrendszeréhez. Mindez pedig jobb munkakörülményeket és végső soron megtakarítást eredményez.

A 2. részben leírjuk, milyen előnyökkel jár, ha a fordítási munkafolyamatot vesszük kiindulási alapul a fordítástámogató eszköz fejlesztésekor. A 3. rész a fájl- és könyvtárkezelés kiiktatásáról szól. A fájl- és könyvtárkezelés véleményünk szerint az a terület, amely a leginkább csökkentheti a hatékonyságot, a meglévő erőforrások fel nem használása révén. A 4. részben a rendszer fejlesztésének lehetőségeit tárgyaljuk. Az 5. rész összegzi a MemoQ alapelveit.

2. A fordítási munkafolyamat kezelése

A fordítás projektszemléletű tevékenység. Az amerikai Projektmenedzsment Intézet *(Project Management Institute)* 2002-es definíciója szerint „a projekt ideiglenes erőfeszítés egy adott cél elérése érdekében. A projektek abban különböznek az üzletviteltől, hogy az üzletvitellel kapcsolatos tevékenységek folytonosak és ismétlődnek, míg a projektek időszakosak és egyediek." A projektnek tehát mindig van kezdete és vége – akkor is, ha egyszerűen csak megszakítják. A fordítási projektek esetében egy vagy több szereplő működik együtt annak érdekében, hogy a forrásdokumentumo(ka)t olyan célnyelvi dokumentummá alakítsa, amely legalább egy nyelvi és kulturális kontextusban megfelelő. A megbízás részletes elemzése után a fordítási projektek különböző tevékenységekre oszthatók, amelyeket más és más személy végezhet.

Robichaud és L'Homme (2003) egyszerűsített listát közöl a munkafolyamat lépéseiről, amely felhasználható a fordítástámogatási és gépi fordítási eszközök használatának tanításához is:

1. A forrásnyelvi szöveg átvétele
2. Első olvasat
3. Terminológiai- és háttérkutatás
4. Fordítás
5. Lektorálás és javítás
6. Átnézés, egységesítés, olvasószerkesztés
7. A célnyelvi szöveg átadása

A munkafolyamat kis számú lépésből épül fel, de a lépések sorrendje változhat, a lépések pedig ismétlődhetnek, így rendkívül sokféle munkafolyamat képzelhető el, főleg a hosszú szövegek fordítása és a szoftverlokalizálás során. Ennek ellenére már a munkafolyamat legelején tudjuk, hogy

(1) kik dolgoznak a projekten,

(2) milyen erőforrásokat használunk fel a projekt során.

A fordítóirodák a legtöbb esetben projektmenedzsert alkalmaznak, akinek a tevékenysége átfogja a teljes munkafolyamatot. A projektmenedzser felelős a minőségbiztosításért, ő biztosítja a munkafolyamat szereplői (fordítók, lektorok, szakértők, kiadványszerkesztők stb.) közötti problémamentes információáramlást, legfőbb feladata pedig, hogy a költségeket alacsonyan tartsa. Ezért a projektmenedzsernek tudnia kell legalább a következőket:

(1) milyen, a projekthez kapcsolható erőforrásokkal rendelkezik a vállalkozás, (mi az, ami „újrahasznosítható")?

(2) hogyan lehet a projektet a lehető legjobban előkészíteni [terminológiakivonatolás, a terminológiai adatbázis előkészítése, stílusútmutató, a fordítási feladatmeghatározás *(translation brief)* írásba foglalása (Fraser 2000)]?

(3) hogyan lehet a munkafolyamatot optimálisan megszervezni?

A projekt során a projektmenedzser biztosítja a zökkenőmentes munkavégzést, a dokumentumok szereplők közti mozgatását, és ő válaszolja meg a fordítók és más szereplők kérdéseit is. A projekt lezárultával a projektmenedzser feladata a dokumentum(ok) eljuttatása a fordítás megrendelőjéhez.

A jelenlegi fordítómemóriák azonban nem teszik lehetővé, hogy a projektmenedzser (a továbbiakban: koordinátor) teljes egészében irányítsa a folyamatokat. A koordinátorok ugyan tudják, hogy a projekt szereplőinek pontosan milyen fájlokra van szükségük, nem tudják ezeket betölteni mások alkalmazásaiba. Tapasztalatból és felmérésekből (pl. Drugan 2004) tudjuk azonban, hogy nem minden fordító rendelkezik elég számítástechnikai tapasztalattal ahhoz, hogy a fájl- és beállításkezelés bonyolult feladatával megbirkózzon. Az egygépes fordítómemóriák első hiányossága itt mutatkozik meg.

Ideális munkakörnyezetben a koordinátorok

(1) erőforrásokat rendelhetnek a felhasználókhoz,

(2) biztosíthatják, hogy a felhasználók hasznosítsák is az erőforrásokat, az összes erőforrást egy csomagba csomagolhatják,

(3) beállíthatják a felhasználók fordítómemóriáit.

Milyen erőforrásokat különböztetünk meg? A szereplők segítségére lehetnek a fordítómemóriák, a terminológiai adatbázisok, az eredeti és a munka közben keletkezett dokumentumok, a beállítások (többek között a hálózati beállítások), a validációs szabályok, az XML DTD-k, de a nyelvi erőforrások is. A munkafolyamat-alapú megközelítés révén minden szereplőnek csak annyit kell tennie, amennyi a feladata. A koordinátor biztosítja a projektszintű hatékonyságot, így csak neki kell magas szintű számítástechnikai ismeretekkel rendelkeznie. A fordítók és a lektorok pedig – ha a koordinátor így szeretné – azonnal dolgozni kezdhetnek, miután megnyitották az *egyetlen* fájlt, amelyet kaptak. A koordinátor nem állíthatja be mindenki fordítómemóriáját, de biztosíthatja, hogy az összes fordító ugyanazokat az erőforrásokat használja. Az erőforrások frissítése, bővítése esetén a fordítók megkapják az új anyagot. Mindez a kö-

vetkezetesség és a hatékonyság növelése révén megkönnyítheti a technikai tudással nem rendelkező fordító dolgát.

Miután minden személy részére létrehoztunk egy alprojektet, elkezdődik a munkafolyamat. A tapasztalat azt mutatja, hogy a munkafolyamat a legtöbb esetben automatizálható. A munkafolyamatot kezelő motornak pedig – ha azt akarjuk, hogy mindenki használja – minél egyszerűbbnek kell lennie.

A kommunikáció legegyszerűbb és legismertebb módja az e-mail. Korábban kifejlesztettünk egy web- és e-mail-alapú munkafolyamat-automatizálási rendszert, a *Fordítás.net*-et. A rendszerben egy robot intézi az összes levelezést, a fájlokat pedig egy kiszolgálón tárolja, és automatikusan továbbítja a folyamatban soron következő személynek. A felhasználó csak válaszol a levélre, csatolja a munkája eredményét tartalmazó fájlt. A robot minden egyebet elvégez – többek között dokumentálja a teljes folyamatot, így biztosítva a megfelelő minőséget.

A munkafolyamatot átfogó rendszert vertikális hálózati komponensnek tekinthetjük, mivel a különböző szerepeket betöltő személyeket köti össze: a koordinátort a fordítóval, a fordítót a lektorral, a koordinátort az ügyféllel stb. Azonban minél jobban automatizáljuk ezt a rendszert, az árajánlattól az elkészült munka átadásáig és a számlázásig – közben persze bármikor lehetőséget adva a koordinátornak a beavatkozásra –, annál inkább hatékony lesz a rendszer, mert

(1) a projekt összes erőforrásának központi webes tárhelye megakadályozza a felhasználóknál bekövetkező adatvesztést,

(2) gyakorlatilag nullára csökken az anyagok továbbításával töltött idő (amikor ember továbbít, a várakozási idő még nagyobb, mint a tranzakcióval töltött idő),

(3) a fájlnevek részleges és a könyvtárak teljes kiiktatása csökkenti a kihagyás lehetőségét.

A horizontális hálózati komponens az egyforma szerepet betöltő felhasználókat köti össze. Ha a fordítás konzisztenciáját és maximális hatékonyságát biztosítani akarjuk, lehetővé kell tenni, hogy a felhasználók egymással folyamatosan kapcsolatot tarthassanak, és minél hamarabb felhasználhassák a munka közben keletkezett erőforrásokat. A horizontális hálózati komponens magja egy tranzakció-naplózásos erőforrás-frissítő. Az ügyfelek (felhasználói gépek) a koordinátor által megadott időközönként szinkronizálják erőforrásaikat a kiszolgálóval. Ha a koordinátor kicsi értéket ad meg, az eredmény kvázi folyamatos erő-forráskövetés.

Látható, hogy ennek a megközelítésnek számos előnye van a kizárólag hálózaton tárolt fordítómemóriával szemben:

(1) nincs szükség folyamatos hálózati kapcsolatra, de akinek van, élhet annak előnyeivel. Magyarországon becslések szerint csak körülbelül a fordítók fele rendelkezik szélessávú internetkapcsolattal, és a helyi hálózaton kommunikáló belső fordítók is kevesen vannak.

(2) a hálózati adatforgalom csökken,

(3) a kiszolgáló számítógép terhelését elosztjuk az ügyfelek között, így egy kiszolgáló több ügyfelet tud kiszolgálni lassulás nélkül.

A kiszolgáló minden ügyfélhez maga kapcsolódik, ezért nem kell beírni a kiszolgáló címét sem, mert a koordinátor állítja be ezt a paramétert. A felhasználók üzeneteket is válthatnak egymással, figyelmeztethetik társaikat a lehetséges problémákra, megjelölhetik a problémás szövegrészeket, terminusokat javasolhatnak és tárgyalhatnak meg (amelyek elfogadás esetén automatikusan be is kerülhetnek a központi adatbázisba), és közvetlenül kommunikálhatnak a koordinátorral.

A horizontális és vertikális hálózati komponensek, az e cikkben nem tárgyalt külső erőforrás-megosztással együtt, optimális teljesítményt biztosítanak hálózati környezetben, és egyszerűsítik a fordítómemóriák használatát.

3. A fájlok és a könyvtárak kiiktatása

A fájlok és a könyvtárak kezelése a fordítók és koordinátorok számára sok gondot okoz. Egy fájl általában az erőforrás egy példányához kapcsolódik, és legfeljebb 255 karakterből álló névvel hivatkozunk rá. A fájlok a fordító számára azonban nem egy, hanem több fontos attribútummal rendelkeznek, és ha az attribútumokat egymás után fűzzük (például egy ilyen elnevezési konvenció alapján: {*forrásnyelvkód*}{*célnyelvkód*}{*tárgykör-kód*}{*ügyfélkód*}{*munkaszám*}{*egyedi azonosító*}) – ez a legjobb, amit tehetünk –, akkor pedig a keresés lesz bonyolult.

A fájlok könyvtárakban találhatók, amelyek vagy egyforma típusú erőforrásokat tartalmaznak, vagy az egy projekthez tartozó erőforrásokat tartalmazzák, a kétféle kategorizáció egyszerre nem lehetséges. Vagy megpróbáljuk megjegyezni, mit hová tettünk, vagy adatbázist készítünk az erőforrásokról, amely minden attribútumot tartalmaz – ehhez jelentős informatikai tudásra van szükség. Esetleg dönthetünk úgy is, hogy megkettőzzük az erőforrásokat, és több tárhelyet használunk fel.

A fájlok és a könyvtárak azért fontosak, mert más szoftverekkel az operációs rendszeren keresztül így lehet érintkezni. Ezért nem lehet tőlük teljesen megszabadulni, de lehetővé tehetjük, hogy a rendszer kezelje őket, és csak egyetlen felületen találkozzunk a fájlrendszerrel.

Az erőforrások természetesen többfélék lehetnek. Minden egyes típushoz meg kell találnunk a leghatékonyabb ábrázolási módot. A fordítómemóriákat és a terminológiai adatbázisokat – amelyeknek természete igen hasonló – lehetséges egyetlen fájlban tárolni. Az ilyen adatbázisok minden egyes bejegyzését el lehet látni többféle attribútummal: ki és mikor készítette a bejegyzést, forrás- és célnyelv, elfogadási státusz, milyen tárgykörbe kerül stb.

A MemoQ-ban a tárgykör *(domain)* kiemelt szerepet kap. A MemoQ a többi fordítómemóriával szemben nem csupán egyetlen tárgykört képes kezelni, hanem az egyes bejegyzéseket többdimenziós térben is el lehet helyezni. Minden tárgykör (pl. autóipar vagy lánctalpas járművek) egy faszerkezet részfája. A különböző tartománydimenziók ortogonálisak, azaz a szövegek különböző szempontok szerinti kategorizálását teszik lehetővé. Annyi tárgykör-dimenziót adhatunk meg, amennyit csak akarunk. Egy nagy fordítóiroda nem csak téma, hanem szervezet (terminológiahasználat stb.), ügyfél (a fordítás megrendelője), és stílus (hivatalos vagy közvetlen stb.) alapján is megadhat tárgyköröket. A tárgykör-információ orientatív, nem pedig restriktív jellegű. Keresés-kor azok a szövegrészek jelennek meg először, amelyeknek több tárgyköre egyezik meg a fordítandó szöveg tárgyköreivel. Azok is megjeleníthetők azonban – más szín-

kódolással –, amelyek teljesen más tárgykörbe esnek, de felszíni hasonlóságot mutatnak a fordítandó szöveggel. Így ahelyett, hogy a fordítómemóriákat fájlokban tárolnánk, csak egyetlen fordítómemóriánk és terminológiai adatbázisunk van, amely viszont tárgykörök és más szempontok szerint szűrhető.

A fordítandó dokumentumok hierarchiáját valamilyen módon meg kell őrizni, mivel a projekt befejeztekor a céldokumentumokat exportálni kell az operációs rendszer „világába". Az információveszteséget minimalizálandó, az egyazon projekthez tartozó dokumentumokat mindig együtt kell kezelni, és ha a fájlokat szétosztjuk, automatikus mechanizmusnak kell visszaállítania az eredeti fájl/könyvtárszerkezetet.

A többi erőforrást, például a DTD-ket, illetve – nyelvi támogatás jelenléte esetén – a szabálybázisokat fájlokként kell kezelni, de a könyvtárstruktúra szükségtelen. A felhasználói felület egyszerűsítése érdekében egyetlen erő-forrásböngészőt kell létrehozni, amelyben a felhasználók (főleg a koordinátorok) láthatják és különböző szempontok alapján szűrhetik az összes erőforrást, és az erőforrásokat projekthez rendelhetik. Így lehet biztosítani, hogy ne veszítsünk el a fordítás során keletkezett értékes erőforrást, és mindent újrahasznosíthassunk.

4. A hatékonyság növelése

A fordítástámogatás e paradigmája szerint minden tevékenységet egyetlen rendszernek kell kezelnie, amelynek fő szervezőeleme a közös munkavégzés. Egyrészt új funkciókra van szükség, másrészt pedig a meglévő funkciókat kell átalakítani.

Az új funkciók beépítésének érdekében testre szabható projekt- és fájlkezelési keretrendszerre és a szabványos erőforrásokat kezelő, lazán kapcsolódó, egymással transzparens módon együttműködő eszközökre van szükség. Ugyanakkor a fordítómemóriák nyelvérzékennyé tétele is rendkívüli mértékben megnövelheti a termelékenységet, javíthatja mind a nyelvi eszköz fedését *(recall)*, mind pedig pontosságát *(precision)*.

Jelenleg a fordítómemóriák *fuzzynak* nevezett algoritmusokkal illesztenek mintákat és tesznek javaslatokat (valójában nem fuzzy logikát alkalmaznak). A morfológiailag egyszerű, többé-kevésbé kötött szórenddel dolgozó nyelvek esetén a statisztika majdnem olyan jól működik, mint az intelligens elemzés, azonban a gazdag morfológiával rendelkező nyelvek esetében egy olyan egyszerű nyelvfüggő művelet, mint a tövesítés is pénzben mérhető jelentős megtakarítást jelenthet. Ha mondatokat is tudunk elemezni, és mondatvázakat létrehozni (Kis-Gröbler-Hodász, 2004), a nyelvtani mintákra is lehet illesztést végezni.

A többdimenziós tartományrendszerrel ellátott intelligens fordítómemória sok kérdést felvet. A felhasználói felületnek egyszerűnek és hatékonynak kell maradnia, mindazonáltal legalább háromféle eredményünk van, amelynek alapján a hasonlóság minőségét értékelni tudjuk. Az első a fuzzy index – a statisztikai hasonlóság –, a második a tartományegyezési index, amely annál nagyobb, minél több tartomány egyezik meg a forrás- és a hasonló szegmens esetében, a harmadik pedig a nyelvi index, amely nyelvi hasonlóságot keres. A fordítót viszont egyetlen dolog érdekli: a legjobb sorrend. Az indexek összegyúrása egyetlen kombinált indexszé nagy kihívást jelent, és további kutatásoknak ad teret.

Az intelligens elemzés használata a fordítási szegmens fogalmát is értelmetlenné teszi. Ha a számítógép elemezni tudja a mondatot, és kisebb egységeket tud elhatárolni,

természetesen nem a mondat az, ami egy szegmensként számít. Akkor mi? Hogyan lehet a szövegek szinkronizálását *(alignment)* megoldani? Hogyan tudjuk biztosítani a kompatibilitást a többi fordítástámogató eszközzel?

5. Összefoglalás: A MemoQ alapelvei

A MemoQ az itt leírt paradigma első implementációja. Megvalósítjuk benne az összes fenti funkciót, és a szoftvert szorosan integráljuk a már működő munkafolyamat-automatizáló rendszerrel, a Fordítás.net-tel. A MemoQ erősen támogatja a közös munkavégzést a hálózati komponensek és az egyszerű, fájlműveletek nélkül működő fordítói felhasználói felület segítségével. Az első nyelvileg támogatott nyelvpár az angol-magyar lesz, amely a MorphoLogic fordítómemória-kutatására és a MorphoLogic kutatói által kifejlesztett MorphoTM fordítómemória-motorra épül. A MemoQ azonban a jelenlegi fordítómemóriákhoz hasonlóan nyelvfüggetlenül is képes működni, hagyományos hasonlósági keresés révén. A MemoQ első verziója 2005 őszére várható.

Elmondhatjuk, hogy nincs átmenet az egygépes alkalmazások és a közös munkavégzést támogató alkalmazások között. A meglévő programokat nem lehet átírni úgy, hogy teljesen megfeleljenek a fenti elvárásoknak. Az új paradigma alapján új fordítómemóriákat kell írni, vagy a régieket teljesen újraírni. Az új megközelítésnek azonban a régi részhalmaza, ezért lehetséges a problémamentes fokozatos áttérés az egygépes megközelítésről a közös munkavégzésre.

6. Köszönetnyilvánítás

A szerzők köszönetet mondanak a MorphoLogic kutatóinak a MorphoTM nyelvérzékeny fordítómemória-technológia rendelkezésre bocsátásáért, illetve az ActiWise Kft.-nek – különösen pedig Cserép Csabának – a Fordítás.net rendszer kifejlesztésében nyújtott segítségéért.

Irodalom

DRUGAN, J. (2004): Training Tomorrow's Translators, Universidad Europea de Madrid. In *Proceedings of Jornadas de Traducción e Interpretación*, Madrid. *(megjelenés előtt)*

FRASER, J. (2000): The broader view: how freelance translators define "translation competence", In: *Schäffner and Adab (eds.), Developing Translation Competence*, John Benjamins, Amsterdam, pp. 51-62

FULFORD, H., GRANELL-ZAFRA, J. (2004): The freelance translator's workstation: an empirical investigation, In: *Proceedings of the Ninth EAMT Workshop*, University of Malta, Valletta, pp. 53-61.

HODÁSZ, G., GRŐBLER, T., KIS, B.: Translation memory as a robust example-based translation system, In: *Proceedings of the Ninth EAMT Workshop*, University of Malta, Valletta (2004) pp. 82-89.

KIS Balázs, LENGYEL István (2003): Új módszerek az emberi fordítás gépi támogatásában. In: *Az I. Magyar Számítógépes Nyelvészeti Konferencia gyűjteményes kötete*, Szegedi Tudományegyetem, Szeged.

REINEKE, D. (2004): Localización y terminología: efectos sinergéticos obtenidos a raíz de un experimento realizado en el aula. Universidad Europea de Madrid, Madrid *(megjelenés előtt)*

RICO PÉREZ, C. (2002): Translation and Project Management. In: *Translation Journal*, 6, 4. http://accurapid.com/journal/22project.htm

ROBICHAUD, B., L'HOMME, M-C. (2003): Teaching the automation of the translation process to future translators, http://www.dlsi.ua.es/t4/docum/robichaud.pdf

Nyelvi hasonlóságon alapuló intelligens keresés fordítómemóriában

Hodász Gábor

Pázmány Péter Katolikus Egyetem
Információs Technológiai Kar
H-1083 Budapest, Práter utca 50/a.
hodasz@morphologic.hu

Kivonat. A cikkben bemutatásra kerül a nyelvi hasonlóság egy definíciója, amely elméleti alapját képezi a nyelvi hasonlóságon alapuló intelligens keresésnek. A definíció lehetőséget ad a hasonlóság Levenshtein-távolság alapú vizsgálatára több nyelvi szinten. Az általános definíción túl bemutatom a fejlesztés alatt álló alkalmazást, amely 3 nyelvi szinten alkalmazza a távolságot: a szavak felszíni alakja szerint, tövesített alakjuk szerint és szófajuk szerint. A példákban így két mondat között egy 3 elemű vektor írja le a hasonlóságot. Bemutatom az elkészült teszt-környezet is, amely szövegfájlban keres egy adott jelölt-mondathoz a definíció szerint hasonló mondatokat. Végül vázolom a további munkákat és terveket.

1 Bevezető

A bemutatott nyelvi hasonlóság definíciójának célja az intelligens keresés megvalósítása a fordítómemóriában. A felhasznált keretrendszer a MorphoLogic Kft-nél fejlesztett MorphoTM fordítómemória, amely a MetaMorpho szabály-alapú fordítástámogató rendszer alapjain, az ott kidolgozott szabály-szintaxist alkalmazza.

A MorphoTM rendszer olyan fordítástámogató eszköz, amelynek célja, hogy a hagyományos fordítómemória-funkciókat nyelvi intelligenciával kiegészítve a jelenlegi rendszereknél többször ajánljon fordítást, és azok jobban közelítsék a kívánt minőségű fordítást. A fordítás egységei a mondatnál kisebb szegmensek (főnévi szerkezetek és az ezeket tartalmazó mondatvázak), amelyeket a forrás- és célnyelvi elemzők állítanak elő. Az adott bemeneti mondathoz hasonló szegmenseket „nyelvi intelligencián" alapuló távolság segítségével keressük, és a megszülető új fordításokat mint szabályokat tároljuk, amelyek a gépi fordítás minőségét folyamatosan javítják.

2 A fordítómemória működésének leírása

2.1 Szabály és minta

A MorphoTM fordítómemória egyesíti a szabály-alapú gépi fordítás és a minta alapú, statisztikus megközelítés előnyeit. A tárolt minták a MorphoLogic MetaMorpho nevű gépi fordítórendszerének formalizmusát követik: az egyes szabályok forrás és célnyelvi részből állnak, azaz minden szabály önállóan tartalmaz egy nyelvi elemet és annak fordítását. A szabályrendszer jellemzője, hogy homogén: nem különböztetünk meg lexikon-szerű és szintaxis-szerű szabályokat [Prószéky96, Prószéky02].

A fenti két tulajdonság lehetővé teszi, hogy a fordítómemóriába kerülő szabályok bármilyen szintű nyelvtani struktúrát leírhatnak, legyen az egyetlen főnév és fordítása, vagy egy mondatváz, amelyben üres helyek jelzik a főnevek helyét és a vonatkozó megkötéseket. Így a szabályok egyben nyelvi minták is, a szabálybázis pedig tekinthető speciális párhuzamos korpusznak is.

A szabályok az elemzés-fordítás folyamán egyszerű unifikációs nyelvtan szerint működnek. Az egyes szabályokban a különböző jegyek (megkötések) határozzák meg a szabály specifikusságát. A szabályban a nem kitöltött megkötések a konkrét mondat elemzése során kerülnek kitöltésre. Így a fordítómemória működése folyamán egy korábban eltárolt minta akkor is releváns lehet az aktuális mondat fordításában, ha előzőleg más morfológiai jegyekkel szerepelt. Ehhez az szükséges, hogy a szabályok eltárolásakor meghatározzuk a kellő megszorításokat, a többi jegyet azonban kitöltetlenül hagyjuk. Így a fordítómemória által megtalált korábbi fordítás csak abban az esetben lesz jelölt, ha kielégíti a szükséges megszorításokat. A nem szükséges megszorítások (pl. szám, személy, idő stb.) pedig a célnyelvi mondat generálása során az aktuális forrásnyelvi megfelelőik szerint kerülnek kitöltésre. Ez a megközelítés lehetővé teszi, hogy a hagyományos fordítómemóriával szemben például az angol 'go' igének különböző idejű alakjait (pl. 'went', 'has gone' stb.) annak ellenére megtalálja a rendszer a minták között, hogy közöttük a karakter-alapú távolság igen nagy.

2.2 Bővítés és fordítás

A MorphoTM alapvetően fordítómemóriaként működik, azaz fejlett eszközökkel támogatja az emberi fordítót, valamint lehetőséget ad az adatbázis bővítésére. A fordítói munka során a fordított mondatok feldolgozásával bővül a szabálybázis, és emellett lehetőség van minták párhuzamos korpuszból való automatikus felvételére is.

A fordítási folyamat során az emberi fordító felügyeli a fordítás folyamatát. A folyamat lépései vázlatosan a következők:

(1) A fordítómemória a berérkező mondatot megelemzi.
   a) Amennyiben ez sikeres, akkor előállnak a forrásnyelvi mondat szavainak lemmái, a mondat váza és a főnévi csoportok (*noun phrase*, NP).
   b) Amennyiben nem sikerül a teljes elemzés, úgy a rendszer megpróbálja egy sekély NP nyelvtannal elemezni a mondatot, így előállítva lemmákat, a vázat és az NP-ket.

c) Ha ez sem jár sikerrel, úgy az egész mondatot váznak tekintjük, amely nem tartalmaz főnévi csoportokat. Ez esetben a morfoszintaktikai elemző előállítja a mondat lemmáit.

(2) A lemmák sorozatát feldolgozzuk a fordítómemória igényeinek megfelően: meghatározzuk a szükséges morfológiai jegyeket, amelyek teljesülését megkívánjuk a tárolt minták közötti keresés során. A feldolgozás során külön kezeljük a speciális lemmákat (pl. Internet vagy e-mail cím, dátum, szám stb.), amelyek az esetlegesen szükséges módosítások után átkerülnek a célnyelvi mondatba.

(3) A főnévi csoportok (NP-k) és a többi lemmából álló mondatváz alapján az „intelligens" kereső hasonló mondatot, illetve hasonló NP-ket keres az adatbázisban.

a) azok a szabályok, amelyek forrásnyelvi oldala teljesen lefedi a mondatot, a mondatvázat vagy egy NP-t, „elsülnek".

b) amennyiben az elsült szabályok nem fedik le a teljes mondatot, úgy a hasonlósági kereső modul a definiált nyelvi hasonlóság szerint hasonló mondatvázakat, illetve NP-ket keres az adatbázisban.

(4) A találatokat megfelelően szűrve és rangsorolva előáll a célnyelvi szegmensfordítás-jelöltek listája.

(5) A jelöltekből, illetve az opcionálisan gépi fordítással előállított célnyelvi szegmensekből összeáll az eredeti mondat felajánlott fordítása, amely tartalmazza a speciális lemmákat is.

(6) A felajánlott fordítás(oka)t a felhasználó elfogadhatja vagy módosíthatja.

(7) A módosított célnyelvi szegmensek elemzése és forrásnyelvi párjukkal való eltárolásuk révén új szabályokkal bővül a fordítómemória. Amennyiben a főnévi szerkezetek szinkronizálása gépi úton nem megvalósítható, úgy a fordító javíthatja a felajánlott hozzárendeléseket.

3 A nyelvi szerkezetek közötti hasonlóság

Ebben a fejezetben leírását adjuk a nyelvi hasonlóságon alapuló keresés egy megközelítésének, amely segítségével a fordítómemóriában levő mondat-vázak és főnévi szerkezetek nyelvi hasonlóság alapján kereshetőek. Ehhez definiáljuk a nyelvi hasonlóság fogalmát.

3.1 Hasonló munkák

Bár a fordítómemóriák és a példa-alapú gépi fordítás elmélete [Nagao84] és alkalmazásai már a kilencvenes években megjelentek, a mondatok közötti hasonlósággal csak az évezredfordulón és utána, azaz napjainkban kezdtek foglalkozni a tudományos munkák. A nyelvi hasonlóság fogalmára azonban egészen a legutóbbi időkig tudtunkkal csak egy kutatócsoport adott definíciót [Mandreoli02]. A gyakorlati alkalmazásra azonban számtalan publikáció született, amelyek tekintélyes része a Levenshtein-távolság [Levenshtein65] kiterjesztésével és a dinamikus programozás eszközeivel ad megoldást a hasonlóság kezelésére [Planas00]. A példa-alapú gépi fordítás megvalósí-

tásához elengedhetetlen a minták hasonlóságának megfelelő kezelése, így a tárgyban született cikkek közül több foglalkozik a témával, illetve részproblémáival, mint a félszabad morfémák (function words) [Sumita93], a főnévi szerkezetek (terminológiák) [Sato93] és az elöljárós szerkezetek [Sumita93].

Az eddigi megközelítések legfontosabb hiányosságai:

- számos megoldás egyedi nyelvi ismereteket kíván, mint pl. az ekvivalenciaosztályok, vagy egyéb szemantikai tulajdonságok
- legtöbbször nem definiálnak nyelvi hasonlósági mértéket, így a találatok halmaza nem rangsorolható
- a legtöbb megközelítésben a mondat a keresés legkisebb egysége

A kereskedelmi forgalomban jelenleg kapható fordítás-támogató rendszerek legtöbbje még a fentebb vázoltaknál is kevesebb nyelvi alapú algoritmust alkalmaz. Hatékonyságukat egyedül gyorsaságuk és a fordítandó szövegek nagyfokú hasonlósága indokolja.

3.2 A nyelvi hasonlósági mértéke

A nyelvi hasonlóság mértékének definiálásakor a következő elvárásokat kell figyelembe venni:

- A hasonlóság mértéke legyen a lehető legnagyobb mértékben független a nyelvi környezettől, a szemantikai kontextustól. A fordítómemória kezeli a kontextust, annak meghatározását a fordítóra bízza, azonban a hasonlóság mértéke ettől független.
- A hasonlóság vegye figyelembe mind a szószintű különbségeket (törlés, beszúrás, csere), mind pedig a morfoszintaktikai jegyek különbségét. A mondatváz és az NP-k megkülönböztetésén kívül szintaktikai jegyeket jelenleg nem veszünk figyelembe.

A teljes mondatok, a mondatvázak és a főnévi csoportok olyan szimbólumok sorának tekinthetők, amelyek meghatározott jegyekkel rendelkeznek, amely jegyek közül van kitöltött és kitöltetlen. Mivel a lexikai jegy csak egy ezen jegyek közül, így minden szegmenst, akár teljes mondat, akár főnévi csoport, olyan szimbólumok sorának tekintjük, ahol a lexikai jegyek kitöltöttek, míg a mondatvázakban a főnévi csoportok helyét lexikai jegy nélküli szimbólumok jelzik, amelyek a fordítás során kitöltődhetnek. Ezért a továbbiakban azonos módon kezelhetünk minden szegmenst.

3.3 A hasonlóság szintjei

A hasonlósági távolság definíciójában a közismert Levenshtein-távolságot (szerkesztési távolság) [Levenshtein65] vesszük alapul. A nyelvi szerkezetek távolságának meghatározásakor figyelembe kell venni a nyelvi szerkezetek különböző szintjeit. Ebben a megközelítésben az m hosszú S szegmenst nem csupán szavak sorozatának tekintjük, hanem a különböző elemzési szinteknek megfelelően L szinten (layer) párhuzamosan összerendelt szimbólumok sorozatának, amely minden szinten m darab szimbólumot tartalmaz. Minden szinten az i-ik szimbólum egyértelműen megfeleltethető az S mondat i-ik szavának, és ebből a szóból különböző szintű nyelvi elemzés útján áll elő.

A MorphoTM rendszerben a következő hasonlósági szinteket definiáljuk:

- felszíni alak *(L1)*
- lemmatizált alak *(L2)*
- szófaj *(L3)*

Példa (1):

The fat mice eat cheese.

	T_1	T_2	T_3	T_4	T_5	T_6
L_1	The	fat	mice	eat	cheese	.
L_2	the	fat	mouse	eat	cheese	Punct
L_3	Det	Adj	Noun	Verb	Noun	Punct

Amennyiben a rendszer elő tudja állítani a mondat vázát és főnévi szerkezeteit, akkor a következő 2 NP- és 1 mondatváz-szegmens születik:

	T_1	T_2	T_3	T_4	T_5	T_6
L_1	The	fat	mice	eat	cheese	.
L_2	the	fat	mouse	eat	cheese	Punct
L_3	Det	Adj	Noun	Verb	Noun	Punct
$k(S_1)$	**NP_1**			**eat**	**NP_2**	

Bár a MorphoTM rendszerben (amint a fenti példákban is látható) a hasonlóságot 3 szinten számítjuk, a definíció lehetővé teszi akár több, akár kevesebb szint alkalmazását is. További szinteken más nyelvi (pl. szemantikai) vagy nem nyelvi (pl. formázási) információ is feldolgozható. Azonban minden egyes szint növelheti a számítás időigényét, amely a nyelvi elemzések többértelműsége, az elemzések bonyolultsága miatt nagyságrendekkel növelheti a feldolgozási időt, valamint további többértelműségi problémákat vethet fel.

Már a fenti példában is, de a mondatok túlnyomó részében a különböző elemzési szinteken fellép a többértelműség problémája. A példamondat 'fat' szava egyaránt lehet melléknév ('kövér, zsíros, hájas'), főnév ('kövérség, zsír, háj') vagy ige ('hízik, hízlal'). A helyes elemzés eldöntése szintaktikai elemzést kíván, amely viszont nincs benne az általunk használt 3 szintben. A szintaktikai elemzés bevezetése, bár elméletileg illeszkedik a vázolt definícióba, azonban olyan bonyolultságú nyelvi problémákat hozna a jelen konkrét megvalósításunkba, amely sem feldolgozási időben sem nyelvi erőforrásban (szabály-bázis méretében) nem megfelelő. A többértelműség kezelésének lehetséges módjait később mutatom be.

4 Levenshtein-távolságon alapuló hasonlóság

4.1 A távolság definíciója

(1) Definíció (Szerkesztési távolság szimbólumok sorozatára). *Legyen S_1 és S_2 két szegmens, amely a következő nyelvi szimbólumokból áll: $\sigma(S_1)=t_1^1..t_n^1$ és $\sigma(S_2)=t_1^2..t_m^2$. A szerkesztési távolság $\sigma(S_1)$ és $\sigma(S_2)$ között $(ed(\sigma(S_1),\sigma(S_2)))$ az a minimális műveletigény (beszúrások, törlések és helyettesítések száma), amely $\sigma(S_1)$-t $\sigma(S_2)$-be viszi.*

A fenti definíció a reprezentáció bármilyen szintjén alkalmazható. Ha a felszíni alakból nyelvi elemzési lépések tetszőleges sorával egy új hasonlósági szint áll elő, akkor az előálló szimbólumok sorát $\phi(S)$-sel jelölve $ed(\phi(S_1), \phi(S_2))$ a fentieknek megfelelően definiálható és számítható. A teljes hasonlóságot az összes hasonlósági szinten kiszámított távolság-értékekből képzett vektorként definiáljuk.

(2) Definíció (Több-rétegű szerkesztési távolság nyelvi szimbólumok sorozatára). *Legyen S_1 és S_2 két szegmens, mely a $\sigma(S_1)=t_1^1..t_n^1$ és $\sigma(S_2)=t_1^2..t_m^2$ nyelvi szimbólumaiból a $\phi_1,...,\phi_i$ nyelvi elemzési lépések segítségével képzett i darab hasonlósági réteget tartalmaz: $L_i,...L_1$. A több-rétegű szerkesztési távolság $\sigma(S_1)$ és $\sigma(S_2)$ között*

$$ED(\sigma(S_1),\sigma(S_2))=$$

$$[ed_{Li}(\sigma(S_{1,Li}),\sigma(S_{2,Li})),ed_{Li-1}(\sigma(S_{1,Li-1}),\sigma(S_{2,Li-1})),...,ed_{L1}(\sigma(S_{1,L1}),\sigma(S_{2,L1}))]$$

A (2) definícióban meghatározott hasonlósági vektor intelligens hasonlóságkeresésre megfelelő, és a későbbiekben látni fogjuk, hogy a hatékonyság növelése érdekében számítási egyszerűsítéseket alkalmazhatunk. Az egyes hasonlósági rétegeken a távolság-érték kiszámításának módja a Levenshtein-távolság számításának hagyományos, dinamikus programozás szerinti algoritmusa [Wagner&Fischer74].

Konkrét alkalmazásunkban a szerkesztési távolság helyett a távolság és a fordítandó mondat hosszának arányával számolunk:

$$d(\sigma(S_1),\sigma(S_2))=\frac{ed(\sigma(S_1),\sigma(S_2))}{|\sigma(S_1)|}$$

Illetve a hasonlósági vektor ennek megfelelően:

$$D(\sigma(S_1),\sigma(S_2))=\left[\frac{ed_{Li}(\sigma(S_{1,Li}),\sigma(S_{2,Li}))}{|\sigma(S_{1,Li})|},...,\frac{ed_{L1}(\sigma(S_{1,L1}),\sigma(S_{2,L1}))}{|\sigma(S_{1,L1})|}\right]$$

Példa (2):

S_1: *The fat mice eat cheese.*

S_2: *Cats eat mice.*

S_1:

	T_1	T_2	T_3	T_4	T_5	T_6
L_1	The	fat	mice	eat	cheese	.
L_2	the	fat	mouse	eat	cheese	Punct
L_3	Det	Adj	Noun	Verb	Noun	Punct

S_2:

	T_1	T_2	T_3	T_4
L_1	Cats	eat	mice	.
L_2	cat	eat	mouse	Punct
L_3	Noun	Verb	Noun	Punct

A teljes mondatokra értelmezett hasonlósági vektor a következőképpen fog alakulni:

$ed_{L3}(\sigma(S_{1,L3}),\sigma(S_{2,L3}))=2$ (2 törlés)

$ed_{L2}(\sigma(S_{1,L2}),\sigma(S_{2,L2}))=4$ (2 törlés és 2 helyettesítés)

$ed_{L1}(\sigma(S_{1,L1}),\sigma(S_{2,L1}))=4$ (2 törlés és 2 helyettesítés)

Így a hasonlósági vektor a következő lesz:

$ED(\sigma(S_1),\sigma(S_2))=[2,4,4]$

A mondatok hosszával osztott hasonlóság:

$D(\sigma(S_1),\sigma(S_2))=[0.33,0.66,0.66]$

Amint az a vektor számításának definíciójából is látszik, az egyes értékek között szoros összefüggés van, azaz a vektor értékei nem függetlenek egymástól. Abban az esetben, ha a szónak nincs többféle elemzése (nincs több szótöve és/vagy szófaja), akkor a felszíni alak egyezése maga után vonja az összes többi réteg egyezését is. Valamint megfordítva: (ugyanebben a meglehetősen ritka esteben) a szófaj-szinten (L_3) való különbözőség szinte biztosan különbözőségre vezet a fentebbi rétegeken.

Amennyiben (és az esetek túlnyomó részében ez az igaz) egy adott szónak több elemzése van, úgy kezelnünk kell a többértelműségből fakadó problémát. Amennyiben rendelkezünk olyan modullal, amely szintaktikai elemzés nélkül, egyéb (pl. statisztikai) úton tud szófaj szinten egyértelműsíteni (POS-tagging), akkor támaszkodhatunk ezen modul által egyértelműsített elemzésre. Amennyiben ilyen modult nem tudunk/akarunk alkalmazni, akkor a keresés algoritmusán kell változtatnunk. Erre két lehetőségünk kínálkozik:

- „vízszintes egyezés-keresés": az adott szegmens összes szavának összes elemzésében megpróbálunk olyan utat találni, amelyik leginkább hasonlít a kereső-mondathoz. A lehetséges utak közül a legkisebb távolságút fogadjuk el a két mondat távolságának az adott szinten.
- „függőleges egyezés-keresés": Az egyes szavakat mintegy különálló egységként kezelve, csak az adott szó lehetséges elemzései között keresünk. Ha van az adott szónak egyező elemzése a kereső mondat adott szavával, akkor elfogadjuk azon a szinten egyezőnek a két szót.

A MetaMorphoTM rendszerben rendelkezésre áll egy szófaji egyértelműsítő (POS-tagger), így a rendszer jelen állapotában a POS-tagger által ajánlott elemzést veszem alapul, a többi elemzést elhagyom. A rendszer következő verziójában már a fent vázolt mindkét egyezés-keresést megvalósítom tesztelés céljából.

A hasonlósági vektor számítási menete mindig az absztrakt szintek felől a felszíni szintek felé (L_3-tól L_1 felé) halad, amely a jövőben beépítendő egyszerűsítő algoritmusokra ad lehetőséget. Amennyiben az absztrakt szinten a távolság túl nagy (adott küszöbérték feletti), úgy nem érdemes a további szinteket vizsgálni.

5 Előnyök

A fenti példából jól látszanak a módszer előnyei:

- a nyelvi elemzés segítségével kiszámolt hasonlóság közel áll ahhoz, amit az emberi fordító is hasonlónak érez (a példában: 'X eat Y.')
- a különböző felszíni alakok közötti eltérések nem befolyásolják a szükségesnél jelentékenyebben a hasonló mondatok megtalálását (pl. közel kerülnek egymáshoz az angol 'go' ige különböző alakjai: 'went', 'has gone' stb.)
- A mondatvázak és a főnévi csoportok külön kezelésével lehetőség nyílik a mondatnál kisebb egységek automatikus fordítására is, még akkor is, ha nem találunk a mondatvázhoz hasonlító mondatot a memóriában.
- A szintek egymás utáni vizsgálatával akár már az első összehasonlításnál (a szófajok vizsgálatánál) eldönthető, hogy folytassa-e az algoritmus az összehasonlítást.

6 A módszer hátrányai és megoldandó feladatai

- Minden olyan algoritmus, amely nyelvi elemzést használ, a következő problémákat hozza be a rendszerbe:
  - a rendszer elveszíti nyelvfüggetlenségét (gazdaságossági következmények)
  - a nyelvi többértelműségek minden szinten megjelennek (ez túlgeneráláshoz vagy hibás találatok tömegéhez vezethet)
  - mind a feldolgozás, mind pedig a keresés idő- és tárigénye többszörösére nő
- A jelenleg angol nyelvre rendelkezésre álló nyelvtanunk túl bonyolult és nagy ahhoz képest, hogy célunk csupán a főnévi szerkezetek és a mondatváz előállítása, sőt, a nyelvészek véleménye sem egységes a tekintetben, hogy mit is nevezhetünk főnévi szerkezetnek. Rendszerünkben ezért egyelőre az emberi fordító feladata lesz a főnévi szerkezetek kijelölése.
- A rendszer nem kezeli az egyes nyelvi szerkezeteket alkotó szavak nyelvi „jelentőségét", azaz ugyanannyira bünteti egy határozószó hiányát, mint az igei szerkezet fejét képező ige különbözőségét, vagy a főnévi szerkezetek esetén egy melléknév különbségét és a szerkezet fejének különbözőségét (pl. a „száraz pezsgő"-től ugyanakkor távolságra van a „félédes pezsgő", mint a „száraz pelenka"). Ennek a megoldása egy későbbi időben várható.
- A keresést meggyorsító indexelési technika kifejlesztése a következő feladat. A rendszer használhatóságának elengedhetetlen feltétele, hogy hatékony indexelési technika támogassa.

7 Összefoglalás

A cikkben bemutattam a nyelvi hasonlóság szerinti intelligens keresésre adott eddigi elméleti megoldásomat, amely a MorphoLogic MetaMorphoTM rendszerébe illeszkedően kerül majd tesztelésre és felhasználásra. Definíciót adtam a több szintű nyelvi hasonlóság mértékére és az ez alapján számítható hasonlóság-vektorra. Valamint vázoltam a közel- és távolabbi jövő kutatási és fejlesztési feladatait.

Irodalomjegyzék

[Levenshtein65] Levenshtein, V. I.: 'Binary codes capable of correcting deletions, insertions and reversals', (1965) Doklady Akademii Nauk, SSSR 163(4) p845-848, also Soviet Physics Doklady 10(8) p707-710.

[Mandreoli02] Mandreoli, F, Martoglia R., and Tiberio, P. (2002) Searching Similar (Sub)Sentences for Example-Based Machine Translation. *Atti del Decimo Convegno Nazionale su Sistemi Evoluti per Basi di Dati (SEBD 2002)*, Isola d'Elba, Italy.

[Nagao84] Nagao, M.: 'A framework of a mechanical translation between Japanese and English by analogy principle', In A. Elithorn and R. Banerji (eds.) (1984), Artificial and human intelligence, 173-180. Amsterdam: North-Holland.

[Navarro01] Navarro, G. 'A Guided Tour to Approximate String Matching.' (2001) ACM Computing Surveys, 33(1):31-88.

[Navarro01] Navarro, G., Baeza-Yates, R., Sutinen, E., Tarhio, J. 'Indexing Methods for Approximate String Matching', (2001) IEEE Data Engineering Bulletin, 24(4), 19--27, Special issue on Managing Text Natively and in DBMSs.

[Planas00] Planas, E., Furuse: O. 'Multi-Level Similar Segment Matching Algorithm for Translation Memories and Example-Based Machine Translation' (2000) COLING-2000, Saarbruecken, Germany, 621-627.

[Prószéky02] Prószéky, G. and L. Tihanyi: 'MetaMorpho: A Pattern-Based Machine Translation Project'. (2002) Translating and the Computer 24, ASLIB, London.

[Prószéky96] Prószéky 'Syntax As Meta-morphology', (1996) Proceedings of COLING-96, Vol.2, 1123–1126. Copenhagen, Denmark.

[Wagner& Fischer74] Wagner, A. R., Fischer M. (1974) The String-to-string Correction Problem. Journal of the ACM, Vol. 21, #1, pp. 168-173.

Iteratív bekezdés- és mondatszinkronizáció

Pohl Gábor

Pázmány Péter Katolikus Egyetem
Információs Technológiai Kar
1083 Budapest, Práter utca 50/A
pohl@morphologic.hu

Kivonat A gépi szövegszinkronizáció (*text alignment*) célja egy forrásnyelvi szövegen és fordításán belül az egymás fordításának tekinthető szövegegységek automatikus összerendelése. Korábbi munkáinkban [5][6] bemutattunk egy hibrid szövegegységhosszakon és statisztikai módszerekkel szűrt horgonyokon alapuló megoldást, amellyel a csak szövegegységhosszakat összehasonlító módszernél jobb bekezdés- és mondatszinkronizációs eredmények érhetők el. Most a gépi szinkronizáció emberi javításának kérdéskörét járjuk körül, valamint bemutatjuk, hogy a szinkronizációs folyamat iteratívvá tételével hogyan csökkenthető az az emberi beavatkozások száma. Az újonan bemutatott módszer nagy szinkronizálási feladatok (például kétnyelvű korpuszok párhuzamosítása) során, illetve fordítómemóriák nagy mennyiségű szinkronizálatlan szöveggel való feltöltése esetén válik különösen hasznossá.

1. Bevezetés

A gépi szövegszinkronizáció (*text alignment*) célja egy forrásnyelvi szövegen és fordításán belül az egymás fordításának tekinthető szövegegységek automatikus összerendelése. Egy szövegpár mondatszintű szinkronizációja lehetővé teszi a szövegpár fordítómemóriába töltését, a fordítás terminológiai konzisztenciájának ellenőrzését, párhuzamos korpuszok létrehozását. Ugyanakkor csupán gépi eszközökkel – a feladat nehézségből adódóan – az esetek többségében nem érhető el teljesen pontos mondatszintű szinkronizáció[1], azaz a szinkronizáció emberi javítása szükséges. Az automatikus szinkronizációt megvalósító megoldások célja az eddigiekben minél tökéletesebb automatikus szinkronizáció előállítása volt, azonban egy szinkronizáló alkalmazás felhasználójának szemszögéből nézve más a cél: minél kevesebb és egyszerűbb emberi beavatkozással tökéletes szinkronizációt létrehozni.

A következőkben a fordítások szinkronizációt megnehezítő tulajdonságait, illetve az egyes szinkronizációs módszerek hibalehetőségeit járjuk körül, majd bemutatjuk a minimális emberi beavatkozás elvét szem előtt tartó iteratív szövegszinkronizáló megoldásunkat.

[1] Szinkronizációnak nevezzük a szövegegységek összerendelésének folyamatát és a folyamat eredményét is.

2. Az automatikus szinkronizáció nehézségei

Az automatikus szinkronizációt megnehezítő tényezők két csoportját különböztethetjük meg. Az egyik csoportba a fordítás, illetve annak szerkesztése során létrejött változtatásokat vehetjük fel, a másik csoportba a különböző nyelvű szövegek automatikus szinkronizációjához szükséges nyelvi előfeldolgozás hibái sorolhatók.

2.1. Változtatások a fordítás és a szöveg szerkesztése során

A fordítók (illetve a fordítás szerkesztői) megváltoztathatják a szöveg szegmentációját, elhagyhatnak, beszúrhatnak, összevonhatnak, szétbonthatnak szövegegységeket (mondatokat, bekezdéseket) a fordítás során. A szerkesztés során – ritka esetekben – az egyes szövegegységek sorrendje is változhat (pl. alfabetikusan rendezett felsorolások).

2.2. Az automatikus nyelvi előfeldolgozás problémái

A bekezdések határai a különböző dokumentumformátumokban általában pontosan rögzítettek; gondot az okoz, hogy az egyes nyelvekhez készített automatikus mondathatár-meghatározó rendszerek többnyire a szövegpár különböző pontjain hibáznak. Ezeket a hibákat a gépi szinkronizáció során – a fordítói szegmenshatár változtatásokhoz hasonlóan – kompenzálni kell.

Horgonykereső módszereknél (lásd később) a horgonyok keresése során esetlegesen alkalmazott (morfológián alapuló) tövesítésnek a különböző nyelvek esetében összevethetőnek kell lennie.

3. Szövegszinkronizációs módszerek és hibalehetőségeik

A szövegszinkronizációs módszerek ismertetésével korábbi munkáinkban [5][6] részletesen foglalkoztunk, most csak hibalehetőségeik szempontjából mutatjuk be röviden az egyes módszertípusokat.

3.1. Szövegegységek hosszát összehasonlító módszerek

A szövegegységhosszakon alapuló módszerek a különböző nyelvű szövegek szövegegységeinek hosszait hasonlítják össze, többnyire Gale és Church valószínűségi modelljére [1] alapozva. Az alkalmazott modell azt feltételezi, hogy a szövegegységek sorrendje nem változik a fordítás során, így ilyen esetekben, illetve beszúrások és elhagyások esetén is, többnyire hibás eredmény várható.

A módszer előnye, hogy egy dinamikus programozás típusú algoritmus alkalmazásával globálisan optimális megoldást keres, többszöri (ismételt) futtatás során mindig azonos eredményre jut. A hibák többnyire lokálisak, hibázás esetén nem a teljes eredmény, csak egyes pontjai lesznek hibásak. Ezzel együtt a hibák

csomósodása is megfigyelhető, a hibát csak újabb – általában közeli ponton megtalálható – hibával tudja korrigálni a globális optimumkereső eljárás.

A szinkronizáció során biztos pontot jelentenek a szinkronizálandó szegmensek végpontjai; a szövegpárt kisebb szegmensekre bontva a módszer megbízhatósága nő, ezért célszerű a szinkronizációt először bekezdésszinten, majd a szinkronizált bekezdéseken belül mondatszinten meghatározni, illetve ezért alkalmazható sikeresen a rövidesen bemutatott iteratív javítást lehetővé tevő módszerünk.

3.2. Horgonykeresés

A horgonykereső eljárások nem törekednek teljes szinkronizációra, azaz a szinkronizálandó szövegpárnak csak egyes kitüntetett pontjait kívánják horgonyokkal összekapcsolni. Horgonyokkal csak akkor lehetne teljes szinkronizációt megvalósítani, ha minden szövegegységet az összes párjával (és lehetőség szerint csak azokkal) horgonyok kötnének össze. Ilyen magas lefedettség (*recall*) és pontosság (*precision*) elérése azonban nem lehetséges.

A horgonyjelöltek statisztikai szűrésére alkalmazott módszerek [7] is feltételezik, hogy a szövegegységek sorrendje nem változik (sokat) a fordítás során. A statisztikai szűrők többnyire kizárják a megváltoztatott pozíciójú szövegegységek horgonyjelöltjeit, ami a horgonyok számának csökkenésével jár. Beszúrások és elhagyások esetében, a horgonyok számának csökkenésével járhat, ha az érintett szövegegység horgonyjelöltet is tartalmaz, mivel csak azok a horgonyjelöltek választhatók ki horgonyként, amelyekből azonos számú szerepel a szövegpár két oldalán.

3.3. Hibrid megoldás

Az általunk korábban bemutatott hibrid, statisztikailag szűrt horgonyokon és szövegegységhosszakon alapuló módszer [6] Gale és Church módszeréhez hasonlóan dinamikus programozásra építve globálisan optimális megoldást keres, amelynek pontosságát a horgonyhasználat növeli[2]. A következőkben azt mutatjuk be, hogyan alkalmazható ez a szinkronizáló motor (vagy más hasonló megoldás), ha a szinkronizáció folyamatát iteratívvá tesszük.

4. Iteratív szövegszinkronizáció

A jelenleg elérhető (többnyire fordítómemória alkalmazások részeként vagy kiegészítőjeként értékesített) szövegszinkronizáló eszközökben a szinkronizáció folyamata többnyire egy automatikus szinkronizációt meghatározó lépésből, majd ennek emberi javításából áll. Egyes eszközökben lehetőség van az utolsó kijavított hiba (vagy tetszőleges más szövegpont) utáni szövegrészek gépi újraszinkronizálására. Ezzel a szinkronizáció folyamatát iteratívvá tették, ugyanakkor

[2] Horgonyok hiányában a módszer Gale és Church módszerével azonos.

bizonyos típusú javítások nehézkesek maradtak (pl. a szövegegységek sorrendjének változásait követő összeköttetések felvétele). A szöveg csak lineáris javítását megengedő módszer előnye, hogy viszonylag kevés fejlesztéssel megvalósítható, ugyanakkor az emberi javítások információtartalma a gépi újraszinkronizálás során ennél jobban is kihasználható.

4.1. Iteratív javítás szegmentáló és kiemelő összerendelésekkel

Az általunk kidolgozott iteratív szinkronizációt lehetővé tevő rendszerben a gépi szinkronizáló kimenetét a korrektor úgy javítja, hogy szinkronizációs összerendeléseket határoz meg, amelyek a következő szinkronizálási ciklusban a gép által megváltoztathatatlan, a szinkronizáció során támpontként használható összerendelések lesznek. A korrektor kétféle összerendelés típust határozhat meg: szegmentálót és nem szegmentálót (kiemelő).

Szegmentáló összerendelés esetén a szinkronizáló program a szövegpárt az összerendelés előtti és utáni szinkronizálandó szegmenspárra bontja, amelyek szinkronizációja kisebb méretük miatt egyszerűbb, illetve gyorsabban megoldható feladat. (A szegmenshatárok a szövegegységhosszakon alapuló módszernél támpontként szolgálnak a szinkronizáció során.)

Nem szegmentáló, azaz kiemelő összerendelések esetén a program az érintett szövegegységeket a szövegből „kiemelve" végzi el a szinkronizációs folyamatot, azaz ilyenkor nem bontja kisebb szinkronizálandó szegmenspárokra a szövegpárt. Ezzel a fordítás során áthelyezett, beszúrt vagy elhagyott szövegrészek okozta szinkronizációs hibák javíthatók. A kiemelő összeköttetéssel megjelölt szövegrészeket a horgonykeresés során is el lehet távolítani a szövegpárból, így horgonyjelöltek hibás felvételét is kiküszöbölhetjük.

4.2. Több hiba javítása egyszerre

Több javító összerendelés is felvehető a szinkronizáló algoritmus újrafuttatása előtt, azonban meg kell tiltani egymást keresztező szegmentáló összerendelések felvételét, illetve fel kell készíteni a rendszert arra az esetre, ha a szövegpár egyik oldalán a szegmentáló összerendelések szövegegységei összeérnek, míg a másik oldalon más (beszúrt vagy elhagyott) szövegegységek ékelődnek közéjük. (Ez utóbbi szövegegységeket ilyenkor beszúrtként vagy elhagyottként kell megjelölni.) Hasonló a helyzet, ha a szinkronizálandó szegmenspár első vagy utolsó szövegegységeit tartalmazza szegmentáló összerendelés.

Az ismertetett javítási módszer előnye, hogy egyes hibák javítása több másik hiba automatikus javítását eredményezheti. Egy hibásan szinkronizált szövegrészben (a hibák csomósodást mutatnak) többnyire egyetlen szegmentáló javítás felvételével helyreállítható a szinkronizáció.

4.3. A szinkronizáló motorral szembeni elvárások

Az előzőekben bemutatott módszer egyszerűnek tűnik, azonban a használt szinkronizáló motorral szemben követelményeket támaszt. A felhasználó ugyanis joggal várhatja el, hogy amit egyszer már jól szinkronizált a program, egy későbbi

iteráció során akkor se rontsa el, ha ő nem rögzítette azt külön (költséges emberi munkával). Ez többnyire teljesül, mivel a kisebb szegmenspárok szinkronizálása egyszerűbb feladat, azonban a teljes sikerhez az is szükséges, hogy a szinkronizáló algoritmus azonos bemenet esetén mindig azonos eredményre jusson. Korábban egyes szerzők (több okból is sikertelenül [5]) próbálkoztak véletlent használó, illetve csak lokális optimumot kereső eljárással [2], ilyen szinkronizáló algoritmus azonban nem alkalmazható iteratívan.

A dinamikus programozást alkalmazó algoritmusok előnye, hogy ha a szinkronizálandó szövegpár egy adott pontján az algoritmus által helyesen felvett párt rögzítünk, és az így kapott kisebb szegmenspárokon elvégezzük a gépi szinkronizációt, akkor az előzővel azonos eredményt fogunk kapni. Tehát az iteratív szinkronizáció során helyes összerendelések rögzítésével az eredményt nem lehet elrontani.

Célszerű, ha az alkalmazott szinkronizáló motor lehetővé teszi virtuális (nulla hosszúságú) szegmensek használatát, amelyekkel a több szegmentáló összerendelés felvétele esetén felmerülő kritikus részek egyszerűen, utófeldolgozási lépések nélkül is szinkronizálhatók. (A beszúrt, illetve elhagyott szegmenseket így a szinkronizáló motor is meghatározhatja, nem szükséges ezeket az eseteket külön kezelni.)

4.4. Horgonyok keresése az iteratív javítás során

Érdekes kérdés, hogy az iteratív szinkronizáció során a résszegmenseken belül vagy a teljes szövegben (természetesen a kiemelt szövegegységeket kihagyva) érdemesebb-e horgonyokat keresni. Nagyobb szövegrészeket tekintve nagyobb a valószínűsége, hogy egy horgonyjelölt a szövegpár egyik oldalán kevesebbszer fordul elő, mint a másik oldalon. Ilyenkor ezek a horgonyjelöltek nem válhatnak horgonyokká. A szövegpár kisebb szinkronizálandó szegmenseiben viszont lehet, hogy horgonyként jelenhetnek meg ezek a jelöltek is, azaz kisebb szegmenspárokat szinkronizálva a teljes szövegpárt tekintve több horgony található. Ugyanakkor az is elmondható, hogy a rövidebb szegmenspárok esetében a horgonyjelöltek száma is kisebb (az egyes szegmenspárokat tekintve), ami csökkenti a hibás horgonyok szűrésére használt statisztikai módszerek megbízhatóságát.

Jelenleg, mivel csak a lehető legpontosabbnak tekintehő horgonyjelölteket válogatjuk ki a szövegpárból, szövegszinkronizációs megoldásunkban a részszegmensenkénti horgonykiválasztást választottuk. Ezt a módszert azonban a későbbiekben lehet, hogy érdemes lesz kiváltani az egész szövegpárt vizsgáló horgonykereséssel, illetve a két lehetőség kombinációja is szóba jöhet. A teljes szövegpár vizsgálata esetén felmerül, hogy a felhasználó által felvett (vagy helyesnek elfogadott) szinkronizációs összerendelések által lefedett szövegegységeken belüli horgonyjelölteket felhasználjuk-e a statisztikai szűrés során, hiszen bár kicsi a valószínűsége, a felhasználói beavatkozás lehet, hogy pont egy hibásan felismert horgony miatt vált szükségessé.

A részszegmenspárok szinkronizálása során a horgonyjelöltek közül kiválasztott horgonyok eredeti horgonyokhoz képesti megváltozása azt jelenti, hogy a dinamikus programozás bemenete is megváltozik, ezáltal veszélyeztetve az eddig

helyesen meghozott szinkronizációs döntéseket. Azonban azt is feltételezhetjük, hogy a kisebb szegmensben több, a szinkronizációt alapvetően segítő horgony fordul elő, és csak nagyon ritkán találkozunk hibásan felvett horgonnyal.

4.5. A szinkronizálási folyamat segítése

Tapasztalataink szerint a korrektor munkáját jelentősen megkönnyíti, ha a szövegpár azon pontjait kiemeljük, amelyeket esetleg helytelenül szinkronizált az automatikus módszer. Ilyen pontok lehetnek a gép által meghatározott (és a korrektor által még el nem fogadott) nem egy-egy típusú összerendelések[3], illetve ezek környezete a szövegpárban. Ezek az összerendelések vagy hibásan jönnek létre, vagy a szövegpár egy kritikus pontjára mutatnak rá.

Hasonlóképpen érdemes kiemelni azokat a szövegegységeket, amelyek gépi szinkronizációja eltér az előző lépésben meghatározottól. Ezek többnyire várható változások, – jobb esetben a hibák eltűnését remélő korrektor pont ezekre a változásokra számít –, ugyanakkor a résszegmenseken belüli eltérő horgonykiválasztás következményei is lehetnek, így célszerű ezekre is felhívni a korrektor figyelmét.

5. Eredmények és további célkitűzések

Az iteratív javítás most ismertetett lehetőségét a MorphoLogicnál fejlesztett szövegszinkronizáló rendszerben valósítottuk meg, és teszteltük, egyelőre csak néhány szövegpáron. Az eddigi módszerek (egyszeri javítás, lineáris javítás és újraszinkronizálás) az új megoldással szimulálhatók, így az eddigieknél rosszabb eredményektől nem kellett tartanunk.

A módszer kiértékelésénél gondot okoz, hogy nem mérhető pontosan, hány javítás szükséges a szinkronizáció elvégzéséhez, hiszen a javítások száma attól is függ, hogy a korrektor hova helyez javító összeköttetéseket.

A sorrendcseréket nem tartalmazó szövegpároknál a javítások száma átlagosan kevesebb, mint fele volt a módszert nem használó egyszeri emberi javítás esetében mérhetőnél. Ez annak köszönhető, hogy egy adott hiba javításához az iteratív megoldásnál többnyire elég volt a hibás összerendelés helyett egyetlen jót megadni, míg egyszeri javítás esetén a hibás összerendelés összes szövegegységének helyes összerendelésekbe rendezéséről gondoskodnia kell a korrektornak.

Csomós hibák javításakor még szembetűnőbb a különbség, ugyanakkor a csomós hibákat már a csupán lineáris javítást és újraszinkronizálást lehetővé tevő szinkronizáló eszközökben is könnyen javítani lehetett. A csak lineáris javítást lehetővé tevő módszerekhez képest lényeges javulás olyan esetekben tapasztalható, ahol a szöveg egyes részeit elmozdították a fordítás során.

Szövegszinkronizáló módszerünket a közeljövőben szeretnénk egy komoly szinkronizálási feladat során is kipróbálni: a SZAK Kiadó informatikai témájú,

[3] Automatikusan 1–1, 1–2, 2–1, 2–2, 0–1, 1–0 összeköttetéseket képes meghatározni a rendszer. Természetesen a korrektor manuálisan másmilyen összeköttetéseket (pl. 3–2) is felvehet.

kétnyelvű (angol-magyar), nyelvenként több, mint 1 millió szavas korpuszát [3][4] kívánjuk szinkronizáló alkalmazásunk segítégével párhuzamosítani.

Hivatkozások

1. Gale, William A.; Kenneth W. Church. A Program for Aligning Sentences in Bilingual Corpora. Computational Linguistics, Volume 19, Number 1, March 1993, Special Issue on Using Large Corpora.
2. Chen, Kuang-hua; Chen, Hsin-Hsi A Part-of-Speech-Based Alignment Algorithm In: *COLING 1994 Volume 1: The 15th International Conference on Computational Linguistics*, 1994.
3. Kis Ádám; Kis Balázs. A Prescriptive Corpus-based Technical Dictionary. In: *Papers in Computational Lexicography: Proceedings of COMPLEX 2003.* Research Institute for Linguistics, Hungarian Academy of Sciences, Budapest, 2003.
4. Kis Balázs; Ugray Gábor. Új korpuszstatisztikai eszköztár kollokációkeresésre. In: *Az I. Magyar Számítógépes Nyelvészeti Konferencia gyűjteményes kötete*, Szegedi Tudományegyetem, Szeged, 2003.
5. Pohl Gábor. Fordítások terminológiai konzisztenciájának vizsgálata. Diplomaterv feladat, Budapesti Műszaki és Gazdaságtudományi Egyetem, Villamosmérnöki és Informatikai Kar, 2003.
6. Pohl Gábor. Szövegszinkronizációs módszerek, hibrid bekezdés- és modatszinkronizációs megoldás. In *Magyar Számítógépes Nyelvészeti Konferencia 2003*, Szegedi Tudományegyetem, Szeged, 2003.
7. Ribeiro, António; Gabriel Lopes; Joao Mexia. Using Confidence Bands for Parallel Texts Alignment In: *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, 2000

IV. Tanulás, felismerés

Teljes mondatszintaxis tanulása és felismerése

Hócza András

Szegedi Tudományegyetem, Informatika Tanszék
6720 Szeged, Árpád tér 2.
hocza@inf.u-szeged.hu
http://www.inf.u-szeged.hu

Kivonat: A dolgozat teljes szintaxis tanulására mutat be egy szabály alapú módszert, amely több mélységű faminták általánosításával oldja meg a problémát. A módszer algoritmusa a korábban hasonló problémákra alkalmazott *RGLearn* egy erre acélra továbbfejlesztett változata. Bemutatásra kerül továbbá egy részfa-kiválasztó módszer ami a részfák alakja alapján működik. Segítségével egy teljes szintaxisfa szintenként lebontható kisebb részfákká. A mondatelemző a tanult szabályrendszer alapján képes felépíteni egy ismeretlen mondat szintaxisát. A megvalósításhoz szükséges információkat a Szeged Korpusz adattárából vettük

Kulcsszavak: teljes szintaxis, gépi tanulás, szabály alapú módszerek

1 Bevezetés

Egy mondat teljes szintaxisának felismerése egy olyan folyamat, amely során meg kell határozni, hogy milyen egymás után következő szavak csoportosíthatók egybe, mint például főnévi, melléknévi, igei, ... szerkezetek. További feladat lehet a mondat egymástól távolabb eső részeinek összekapcsolása, azaz a vonzatkeretek meghatározása. Ezek az információk nélkülözhetetlenek a gépi eszközökkel történő mondatértés megvalósításánál. Ez azt jelenti, hogy be kell azonosítani a mondat minden egyes szavát, szócsoportját és mivel ezek egymásra épülnek, fel kell tárni a mondatszintaxis faszerkezetét. További jellemzője még a teljes szintaxisnak, hogy a végeredményül kapott szintaxis-fa összefüggő, teljesen lefedi a mondatot és a gyökere hagyományosan egy S szimbólum.

Az elkészült mondatszintaxis számos természetes nyelvi feladathoz szolgálhat nélkülözhetetlen információkkal. Ilyen feladat lehet például az adatbányászat, információkinyerés, gépi fordítás. Ezekből a lehetőségekből az üzleti hírekből történő információkinyerés megvalósítása vált számunkra a fő célkitűzéssé. Az általunk kifejlesztett automatikus információkinyerést (Hócza et al., 2003) megvalósító programlánc (ToolChain) moduljai folyamatos továbbfejlesztés alatt állnak. Ezek a modulok különféle természetes nyelvi feladatokat oldanak meg, mint például mondat- és szószegmentálás, morfológiai elemzés, szófaji egyértelműsítés, szintaxis felismerés, ontológiai elemzés és szemantikus keretek illesztése.

A különféle gépi tanulással megvalósított természetes nyelvi feladatokhoz szükséges információk a Szeged Korpusz (Alexin et al., 2003) adattárából származtak. Az 1.2 millió szavas korpusz különböző (iskolai, szépirodalomi, számítógépes, jogi, üzleti) szövegtípusokra tartalmazza a nyelvész szakértők által megvalósított szófaji egyértelműsítést és teljes szintaxis elemzést. A teljes szintaxis előállítása két lépésben történt, a munka első fázisában a főnévi szerkezetek (NP) bejelölése történt meg, a teljes szintaxis bejelölésének a munkálatai nemrég fejeződtek be.

A dolgozat a következő módon épül fel: a 2. rész általánosan mutatja be a mondatszintaxis kutatását, a 3. részben a tanulási példák előállításáról lesz szó, a 4. rész az alkalmazott gépi tanulási módszert írja le, az 5. rész a szabályok segítségével történő szintaxis felismerésről szól és végül a 6. rész összefoglalja az elért eredményeket.

2 A mondatszintaxis tanulása és felismerése

A magyar nyelv számos olyan elemet tartalmaz, ami megnehezíti a szintaxis felismerést más indoeurópai nyelvekhez képest. Az egyik nagy probléma a viszonylag szabad szórend, azaz hogy egy adott mondat szavai, mondatrészei sokféle sorrendben átrendezhetők anélkül, hogy megváltozna annak értelme. Ennek a ténye jelentősen megnöveli a lehetséges minták, nyelvi sémák számát, ami rontja az ezen alapuló gépi tanulás hatékonyságát. Ezt a hatást még fokozza számos egyéb nyelvi sajátosság, mint például az, hogy a főnévi csoportokból hiányozhat a névelő és akár maga a főnév is melyek jól felismerhető határelemei lehetnének ezeknek a csoportoknak. Mindezeket a magyar nyelv ragozással, képzők alkalmazásával oldja meg, ami viszont bonyolultabbá teszi a morfológiai elemzést és a szófaji egyértelműsítést, melynek hibái továbbgyűrűzhetnek az ezeken alapuló mondatszintaxis felismeréséhez.

A mondatszintaxis felismerésének létezik egy olyan megközelítése is, amely csak részleges elemzést (*Shallow Parsing*) végez el. Ez a módszer, előállítva a mondatszintaxis leglényegesebb elemét, a főnévi csoportokat, igen hasznos információt nyújt az erre épülő feladatoknak, például az információkinyerésnek.

Az eredményeket az összehasonlíthatóság érdekében közös mérőszámokkal kell jellemezni. Erre a gyakorlatban a következő 3 érték szolgál:

- Pontosság: a helyesen felismert szócsoportok száma / az összes felismert szócsoportok száma.
- Fedés: a helyesen felismert szócsoportok száma / a teszt mintában ténylegesen szereplő szócsoportok száma.
- Középarány ($F_{\beta=1}$): 2* Pontosság* Fedés/(Pontosság+ Fedés)

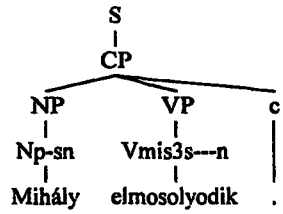
Angol nyelvre számos eredmény létezik a mondatszintaxis felismerésének témakörében. Az első publikációban (Abney, 1991) nyelvtani kódok alapján osztályozta a szavakat, hogy azok kezdő, vég vagy belső elemei-e egy adott típusú frázisnak. A Penn Treebank (Marcus et al., 1993) annotált szövegeiben elkülönítettek egy részt, amely a megjelenése óta összehasonlítási alapot képez a témához kapcsolódó publikációk eredményeihez. (Ramshaw and Marcus, 1995) transzformáción alapuló tanulást valósított meg ($F_{\beta=1}$=92.0). (Argamon, 1998) egyszerre végezte főnévi és igei szerkezeteket felismerését ($F_{\beta=1}$=91.6). (Tjong Kim Sang and Veenstra, 1999) bevezette a több fokozatban (kaszkád) alkalmazott felismerést ($F_{\beta=1}$=92.37). A legújabb módszerek úgy érnek el javulást az eredményekben, hogy több módszert is összekombinálva,

szavazással hozzák meg a döntéseket, (Tjong Kim Sang, 2000) öt különféle módszert kombinált össze ($F_{\beta=1}$=93.26).

Magyar nyelvre idáig nem készült igazán jó minőségű szintaxiselemző program. Az igazi probléma nem is a felismerő program megvalósításában, hanem az azt működtető szabályrendszer előállításában van. Az előzőekben vázolt nehézségek miatt szinte lehetetlen olyan nyelvész szakértők által kézzel készített szabályrendszert megalkotni, ami megfelelő hatékonyságú és minden lehetséges esetre kiterjed. A másik probléma, hogy idáig nem volt elegendő mennyiségű annotált magyar szöveget tartalmazó korpusz, ami a gépi módszerek alkalmazását lehetővé tette volna. A MorphoLogic által kifejlesztett HumorESK mondatelemző (Kis, 2003) 1995 óta folyamatosan fejlődik. Ez idő alatt különféle nyelvészeti területeken alkalmazták, mint például tulajdonnév-, főnévi csoport- és igevonzat-felismerés. Fő jellemzője, hogy a szimbólumokhoz jegyszerkezeteket (*feature structure*) kapcsol és elemzési erdőt épít az egyes jegyek öröklötetésével. Az elemzőben használt nyelvtan nyelvész szakértők közreműködésével állt elő. A Nyelvtudományi Intézet készülő szintaktikai elemzője (Váradi, 2003) főnévi szerkezeteket ismer fel, amely (Abney, 1998) ötletén alapulva reguláris kifejezésekkel leírt, több fokozatú (kaszkád) szabályrendszert alkalmaz. A szabályrendszert nyelvész szakértők készítették a CLaRK rendszer (Simov, 2001) segítségével és az elemzések futtatása is ezzel történt. Az elemző tesztjét egy kisebb annotált szövegen végezték ($F_{\beta=1}$=58.78). A Szegedi Egyetemen a nyelvtan előállítása gépi tanulási módszerekkel történik, egy ilyen nyelvtanon alapuló elemző (Hócza, 2004) szintén főnévi szerkezetek felismerésére készült. Az ebben alkalmazott reguláris kifejezésekkel leírt, környezetfüggetlen, valószínűségi nyelvtan tréningjének forrását, a Szeged Korpusz annotált szövegei adták. A kiértékelés a korpusz teszt célra elkülönített szövegein készült (általános szövegek: $F_{\beta=1}$=78.59, üzleti hírek: $F_{\beta=1}$=83.11).

3 A tanulási példák előállítása

A tanulási fázis megkezdése előtt az XML fájlokban tárolt információkat át kell alakítani gépi tanulásra alkalmas formába, azaz a rendelkezésre álló információkat úgy kell átszervezni, hogy az tanulási probléma legyen. A mondatban szereplő szócsoportok egymásba ágyazottak, ami azt jelenti, hogy ezek a szerkezetek fa-struktúrát alkotnak. Ezért a gépi tanulás megvalósításához a fa-struktúrát le kell bontani, olyan mintákat elkülönítve benne, melyeket alkalmazni lehet egy tetszőleges (ismeretlen) mondat fa-struktúrájának az előállítására. Egy szócsoport megadása az elemek nyitó és záró címkék közötti felsorolásából áll. A szavakhoz tárolt információ MSD (Morpho-Syntactic Description) kódból (Erjavec and Monachini, 1997) és a szótőből áll. Az annotált szövegek a következő csoportokat tartalmazhatják: főnévi csoport (NP), melléknévi csoport (ADJP), határozószói csoport (ADVP), igei csoport (VP), főnévi igenév (INF), tagadószó (NEG), igekötő (PREVERB), kötőszó (C), névmás (PP), tagmondat (CP), mondat(S).



A mondat leírása:
<S> <CP> <NP> Np-sn|Mihály </NP> <VP> Vmis3s---n|elmosolyodik </VP> cl. </CP> </S>

Kinyerhető minták:
<NP> Np-sn|Mihály </NP>
<VP> Vmis3s---n|elmosolyodik </VP>
<CP> NP VP cl. </CP>
<S> CP </S>

1. Ábra: Egy rövid példa mondat teljes szintaxisra és az ebből kinyerhető mintákra.

Az 1. ábrán egy lehetséges példa látható minták kinyerésére, mely több lépésben áll elő, úgy hogy mindig csak azok a csoportok kerülnek kiírásra, melyben a nyitó és záró címkék között csak terminálisok fordulnak elő. Ennek az a hátránya, hogy mivel a minták környezetfüggetlenek elvesznek a minta környezetében szereplő információk, melyek meghatározzák, hogy milyen körülmények között alkalmazható a minta ismeretlen szövegen. Például a 2. ábra mintáiból kinyerhető az **NP → Np-sn|Mihály** szabályról azt lehet elmondani, hogy a Mihály szót megelőzheti névelő és melléknév (a szőke Mihály), tehát ilyen esetben hibát okoz az alkalmazása. Ennek a problémának a mérséklésére a mintagyűjtő algoritmus több egymásba ágyazott szócsoportot is kigyűjt, azaz nem csupán terminális-sorozatok, hanem részfák lesznek eltárolva. Azonban a másik véglet sem túl jó, a túl nagy részfák tárolása, mert ezek többnyire annyira egyediek, hogy nem fordulnak elő még egyszer a tréning vagy a teszt szövegben. A középutas megoldást két faalak típus alkalmazása adja:

1. "Gödör": Legalább 2 terminálist tartalmaz, egy monoton mélyülő és egy monoton emelkedő szakaszra osztható, csak egy irányváltás van benne. Ez egy rekurzív szerkezet, ami többnyire úgy keletkezik, hogy a belső fához, hozzácsapódik 1-2 terminális egy-egy újabb fa-szintet alkotva, például:

```
<NP>
    <NP>
        Tilegy
        <ADJP>
            Afp-sn|rettenetes
        </ADJP>
        Nc-sn|örvény
    </NP>
    Nc-sp---s3|szélén
</NP>
```

2. "Füzér": Több irányváltás is lehet benne, de a lebontatlan részfák max. 1 mélységűek és 1 hosszúak. Ezek vagy felsorolás jellegű szerkezetek, vagy a már lebontott részfák összefűzései egy szerkezetbe, például:

```
<NP>
    <NP>
        Np-snlMihály
    </NP>
    <C>
        Ccswlés
    </C>
    <NP>
        Np-snlErzsi
    </NP>
</NP>
```

Ezek a faalak típusok elég gyakran előfordulnak az annotált szövegekben, több kisebb részfát magukba foglalnak, nem túl hosszúak és nem is túl rövidek, valamint belátható, hogy segítségükkel tetszőleges szintaxis-fa lebontható részfákra.

4 Minták tanulása

A tanulási fázis az előző fejezetben vázolt előfeldolgozással előállított minták általánosítását végzi el. Az általánosítás azt jelenti, hogy a minta szavaihoz tartozó információk közül elhagyjuk a szótőt vagy az MSD kód egyes betűit. (MSD kód minden betűje valamilyen morfológiai információt jelöl.) Az általánosítással az adott minta több esetre alkalmazható is lesz és ez a kulcsa annak, hogy az előállított minták ismeretlen szövegre is alkalmazhatók legyenek. A túlzott általánosításnak viszont az a hátránya, hogy a minta több olyan esetet is lefedhet, melyet nem kellene, azaz megnőhet a hibás felismerések száma. Ennek elkerülésére a tanuló algoritmus optimalizálja az általánosítást, hogy a lehető legkisebb hiba mellett legyen megtanult minta halmaz a legáltalánosabb. Ez úgy valósul meg, hogy minden új minta előállításakor a tanuló algoritmus hiba-statisztikát készít a tréning mondatokon.

A fentebb vázlatosan leírt módszer egy többszörösen átdolgozott saját fejlesztésű algoritmus, az RGLearn (Hócza et al., 2003). Ennek az algoritmusnak egy változata volt alkalmazva a főnévi szerkezetek felismerésére (Hócza, 2004), mely most újabb továbbfejlesztésen esett át a teljes szintaxis felismerése kapcsán. Az algoritmus vázlatos működése következő:

```
amíg van feldolgozatlan minta addig {
    (1) vesz egy új mintát
    (2) előállítja a minta a legáltalánosabb alakját (a
          szavak MSD kódjának csak az első betűjét, azaz
          a szófajt hagyja meg)
    (3) előállítja a pozitív (helyes) fedéseket
    legyen v=0 amíg van javulás addig v=v+1 {
          (4) előállítja az összes lehetséges szabályt, ami
                legfeljebb v számú attribútumot hoz be
                a pozitív fedések adataiból
          (5) hibastatisztika készítése a szabályokra
          (6) a legjobb szabályokkal lefedi a pozitív
                fedéseket
    }
    (7) a legjobb fedésben résztvevő szabályokat eltárolja
}
```

Megjegyzések az egyes pontokkal kapcsolatban:

(1) A minták az előfeldolgozásból származnak, a 3. fejezetben leírtaknak megfelelően a tréning mondatok szintaxis fáinak lebontásával. A minták lebontási szintenként vannak csoportosítva, azaz, hogy hányadik lebontási menetben sikerült kinyerni az adott mintát. Ez az információ azért fontos, mert a tesztmondaton is ilyen sorrendben kell majd alkalmazni a mintákból kapott szabályokat.

(2) A minta legáltalánosabb alakja a nyitó és záró címkékből (pl.: <NP>, </NP>), valamint a szavak MSD kódjainak első betűjéből (azaz a szófajból) áll, szótövek nélkül.

(3) A pozitív fedések halmaza úgy áll elő, hogy a tréning mondatokból kigyűjtjük azokat a részsorozatokat, melyekre az általánosított minta hibátlanul ráilleszthető.

(4) Egy adott szabály úgy áll elő, mint az általános minta specializációja, hogy veszünk egy sorozatot a pozitív fedések halmazából és bizonyos számú attribútumot (betűt az MSD kódból, vagy a szótövet) behozunk belőle, kiegészítve vele a legáltalánosabb mintát. Ez a lépést az összes lehetséges módon megteszszük. A behozott attribútumok száma 0-ról indul és addig növeljük amíg javulást tapasztalunk a (6)-os pont elvégzése után, azaz a hibás fedések száma csökken.

(5) Minden szabályt kiértékelünk megvizsgálva azt, hogy hány esetben lehet helyesen vagy esetleg hibásan ráilleszteni a tréning mondatokra, ezekből az adatokból egy $F_{\beta=1}$ értéket képezünk.

(6) Kiválasztjuk a szabályoknak azt a részhalmazát, ami a legkevesebb hibával fedi le a (3) pontban kigyűjtött jó esetek halmazát teljes mértékben.

(7) Amikor már nincs javulás, vesszük a legjobb (6)-os pontban kigyűjtött szabályrészhalmazt és ennek szabályait eltároljuk.

5 Szintaxis felismerés

A szintaxis felismerésnél az előfeldolgozás fordítottja, a szerkezetek felépítése történik. Akkor jó az elemző, ha a tréning példákon nagy pontossággal reprodukálni tudja a megtanult szerkezeteket és ismeretlen teszt szövegen is jó hatásfokkal, az etalonnal egyezőnek ismer fel. A tanuló algoritmus által előállított szabályrendszer tesztelése az alábbiakban ismertetett algoritmussal történt, melynek előnye az, hogy egyértelműen előállít egy nagy valószínűséggel jó struktúrát egy adott mondatra. Azonban van hátránya is, például nem biztos, hogy a legjobbat, mivel nem az összes lehetséges fa közül választ, hanem minden választási ponton döntést hoz arról, hogyan menjen tovább. Egy adott mondat teljes szintaxisának előállítása a következő algoritmussal történt:

```
amíg van alkalmazható szabály addig {
     az osszes terminálisra {
     az osszes szabályra {
     ha a szabály illeszthető a terminális poziciótól akkor {
            (1) a terminális megjelölése a szabállyal
     }}}
     az osszes megjelölt terminálisra {
     ha nem része más megjelölt terminálisnak akkor {
            (2) a behelyettesítés elvégzése
     }}}
}
```

Megjegyzések az egyes pontokkal kapcsolatban:

(1) A terminális megjelölése azt jelenti, hogy hozzárendeljük a terminálishoz a szabály behelyettesítéshez szükséges adatokat, mint pl. a szabály azonosítója, pontértéke ($F_{\beta=1}$) az illeszkedő sorozat utolsó terminálisa. Ha a terminális már megjelölt, ha az új szabály pontértéke jobb, akkor lecseréljük erre.

(2) A behelyettesítés azt jelenti, hogy a sorozatot egyetlen terminális szimbólummal (az illeszkedő részfa gyökere) helyettesítjük. Egy terminális akkor része egy másik megjelölésnek, ha a szóban forgó terminális sorozat eleje és vége között van.

A tesztelés tréningje úgy történt, hogy a Szeged Korpusz adattárából véletlenszerűen kiválasztott 1000 mondatra lett lefuttatva a 4. fejezetben ismertetett tanuló algoritmus. Az így kialakult szabályrendszer tesztje pedig újabb 100 véletlenszerűen kiválasztott mondaton történt. A teszt eredményét az alábbi táblázat tartalmazza:

Tényleges szócsopotok száma	1835
Felismert szócsopotok száma	1846
Helyesen felismert szócsopotok száma	1557
Pontosság	84,34%
Fedés	84,85%
Középarány ($F_{\beta=1}$)	84,59%

2. Ábra: Teszteredmények 100 mondatra.

A kialakult szabályok pontértéke alapján a legrosszabbak a tagmondatok felismerésére tanult szabályok. A minták a tagmondatok szintjén túlságosan egyediek, kevésnek tűnik a rendelkezésre álló információ, ami arra utal, hogy az elemzést előállító nyelvész szakértők valószínűleg szemantikai információkat is figyelembe vettek a munkájuk során.

6 Összefoglalás és fejlesztési lehetőségek

A dolgozatban bemutatásra került egy teljes szintaxis felismerésére alkalmazott szabály alapú tanuló módszer. Az előállított szabályrendszer segítségével egy elemző algoritmus azonnali döntések meghozatalával a legvalószínűbbnek ítélt szintaxis fát építi fel egy tetszőleges ismeretlen mondaton. A teszteredmények biztatóak, egy nagyobb anyagon végzett tréning során kialakuló szabályrendszer alkalmas lehet az információkinyerést támogató teljes szintaxis felismerést végző modul megvalósításához.

A közeljövőben szeretnénk kifejleszteni egy olyan elemzőt, amely képes az összes lehetséges elemzés előállítására és ezek közül a legjobb kiválasztására. Tervezzük ontológiai információk figyelembevételét is az elemzések során.

Irodalom

Abney S. (1991) Parsing by chunks, in *Principle-Based Parsing*. Kluwer Academic Publishers.

Abney S. (1996) Partial Parsing via Finite-State Cascades, in *Proceedings of ESSLLI'96 Robust Parsing Workshop*, pp. 1-8.

Argamon, S., Dagan, I., and Krymolowski, Y. (1998) A memory-based approach to learning shallow natural language patterns, in *Proceedings of 36th Annual Meeting of the Association for Computational Linguistics (ACL)*, Montreal, pp. 67-73.

Alexin, Z., Csirik, J., Gyimóthy, T., Bibok, K., Hatvani, Cs., Prószéky, G., Tihanyi, L. (2003) Manually Annotated Hungarian Corpus, in *Proceedings of the Research Note Sessions of the 10th Conference of the European Chapter of the Association for Computational Linguistics EACL03*, Budapest, Hungary, pp. 53–56.

Erjavec, T. and Monachini, M., ed. (1997) Specification and Notation for Lexicon Encoding, *Copernicus project 106 "MULTEXT-EAST"*, Work Package WP1 – Task 1.1 Deliverable D1.1F.

Hócza, A., Alexin, Z., Csendes, D., Csirik, J., Gyimóthy, T. (2003) Application of ILP methods in different natural language processing phases for information extraction from Hungarian texts, in *Proceedings of the Kalmár Workshop on Logic and Computer Science*, Szeged, Hungary, 1-2 October, pp. 107-116.

Hócza (2004) Noun Phrase Recognition with Tree Patterns, in *Proceedings of the Acta Cybernetica*, Szeged, Hungary

Kis, B., Naszódy, M., Prószéki, G. (2003) Komplex (magyar) szintaktikai elemző rendszer mint beágyazott rendszer, *MSZNY 2003 konferencia kiadványa*, Szeged, 145-151 oldal.

Kuba, A., Bakota, T., Hócza, A., Oravecz, Cs. (2003) A magyar nyelv néhány szófaji elemzőjének összevetése, *MSZNY 2003 konferencia kiadványa*, Szeged, 16-23 oldal.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993) Building a large annotated corpus of English: the Penn Treebank, Association for Computational Linguistics.

Muggleton, S. and Feng, C. (1992) Efficient Induction of Logic Programs, in Inductive Logic Programming (ed.: S. Muggleton), *Academic Press*, New York, pp. 281–297.

Plotkin, G.D (1970) A note on inductive generalization, *Machine Intelligence* (eds: B. Meltzer and D. Michie), Vol 5.

Ramshaw, L. A., and Marcus, M. P. (1995) Text Chunking Using Transformational-Based Learning, in *Proceedings of the Third ACL Workshop on Very Large Corpora*, Association for Computational Linguistics.

Simov K. (2001) CLaRK – an XML-based System for Corpora Development, in *Proceedings of the Corpus Linguistics 2001 Conference*, Lancaster, pp. 553-560.

Tjong Kim Sang, E. F., and Veenstra, J. (1999) Representing text chunks, in *Proceedings of EACL '99*, Association for Computational Linguistics.

Tjong Kim Sang, E. F. (2000) Noun Phrase Recognition by System Combination, in *Proceedings of the first conference on North American chapter of the Association for Computational Linguistics*, Seattle, pp. 50-55.

Váradi, T. (2002) The Hungarian National Corpus, in *Proceedings of the Second International Conference on Language Resources and Evaluation LREC2002*, Las Palmas de Gran Canaria, pp. 385–389.

Váradi T. (2003) Shallow Parsing of Hungarian Business News, in *Proceedings of the Corpus Linguistics 2003 Conference*, Lancaster, pp. 845-851.

Statisztikai alapú tulajdonnév-felismerő magyar nyelvre

Farkas Richárd[1], Szarvas György[1]

[1] MTA-SZTE, Mesterséges Intelligencia Tanszéki Kutatócsoport,
6720 Szeged, Aradi vértanúk tere 1., Hungary,
{rfarkas, szarvas}@inf.u-szeged.hu

Kivonat: Ebben a cikkben bemutatunk egy döntési fa alapú statisztikai tulajdonnév-felismerő rendszert magyar nyelvre. A modellt a Szeged Korpusznak az MTI honlapjáról származó, gazdasági rövidhíreket tartalmazó szegmensén tanítottuk és teszteltük, s vizsgáltuk annak pontosságát különböző méretű és összetételű tanuló halmazok felhasználása esetén. A feladathoz csak numerikusan kódolható információkat használtunk fel (nem használtuk fel a szóalakot), melyek között előfordultak speciálisan a magyar nyelv tulajdonneveinek helyesírására vonatkozó előírásai is, de a feladat során célunk volt a gazdasági hírekben előforduló, nagy számú idegen eredetű tulajdonnév azonosítása is. A kísérletek során legjobb pontosságot mutató modell 89,6%-os F mértéket ért el.

1 Bevezetés

A tulajdonnevek azonosítása (és kategorizálása) a folyó szövegben meghatározó fontosságú számos számítógépes nyelvfeldolgozó alkalmazás során. Példaként tekinthetjük a különböző információkinyerő rendszereket, ahol a tulajdonnevek általában jelentős információt hordozó szerepet töltenek be a szövegben, vagy a gépi fordítási alkalmazásokat, ahol értelemszerűen más módon kell kezelni emberek, szervezetek neveit, mint a szöveg többi részét.

Számos más nyelven eredményesen alkalmaztak különböző gépi tanulási eljárásokat tulajdonnév-felismerésre, sőt sok esetben ezek az eljárások egyszerre több célnyelven is hatékonynak bizonyultak [1]. Magyar nyelvre is készült már alkalmazás, a MorphoLogic Kft. HumorESK [6] nyelvi elemzője nyelvészek által összeállított szakértői szabályokon alapul.

E cikk célja egy statisztikai tulajdonnév-felismerő rendszer bemutatása, mely meglehetősen jó pontossági mutatóival igazolja, hogy más nyelvekhez hasonlóan magyar szövegekben is eredményesen alkalmazhatók tanuló modellek tulajdonnevek felismerésére. A feladatra a C4.5 [7] döntésifa-tanuló algoritmust használtunk, mert a fa struktúrájában kódolt döntések könnyen értelmezhetők a felhasználó által, valamint az egyes leveleken keletkező osztályozások jól mérhető pontossági adatokkal szolgálnak arra nézve, melyek azok a szegmensei a tulajdonnevek sokaságának, ahol további vizsgálatokkal a modell finomításra szorulhat.

A következő fejezetben (2) röviden ismertetjük a tulajdonnév-felismerési problémát, majd az általunk felállított tanuló modell bemutatása következik (3). Ezt követően ismertetjük az elvégzett kísérletek eredményeit (4), valamint röviden értékeljük azokat, felsorolva a modellel kapcsolatos további teendőket (5).

2 Tulajdonnév felismerés

A tulajdonnevek felismerése tekinthető egy taggelési feladatként, ahol minden szóra (tokenre) egy folyó szövegben a cél: megjelölni, hogy az adott nyelvi elem része-e egy tulajdonnévnek, és ha igen, akkor milyen kategóriába sorolható.

Kísérleteink során a CoNLL konferenciákon [1] is használt <helységnév, személynév, szervezet, egyéb> kategóriákat használtuk, egyrészt a jobb összemérhetőség végett, másrészt ez a 4-es osztályozás jól illeszkedik az információkinyerési alkalmazások céljaihoz, melybe a modellt beépíteni akarjuk. Az egyetlen lényegi eltérés az ott definiált osztályozáshoz képest, hogy az <egyéb> kategóriába angol és más nyelveken általában beleveszik a különböző, mennyiségeket jelölő kifejezéseket is, míg mi ezeket kihagytuk a modellből (egyrészt mert ezek a magyar nyelvben általában nem minősülnek tulajdonnévnek, másrészt ezek azonosításával a Szegedi NLP csoportban egy másik modul foglalkozik [5]).

Legtöbbször megkülönböztetik az osztályozás során a tulajdonnevek kezdő tokenjeit, és a tulajdonnév részét képező belső szóelemeket. Ennek elsősorban akkor van jelentősége, amikor a szövegben egymást követően több, azonos kategóriába tartozó tulajdonnév található, mert ilyenkor ezek segítségével állapítható meg azok kezdőpozíciója. A mi esetünkben ettől a megkülönböztetéstől eltekintettünk, azaz célunk csak annak eldöntése volt, hogy az adott token része-e tulajdonnévnek, vagy sem. A későbbiekben természetesen akár egy szakértői szabályrendszerrel, akár a tanuló mo-dellbe való beépítéssel szükséges lesz a szókezdő tokenek azonosítása is.

A tanuláshoz a Szeged Korpusz gazdasági hírekből álló részét használtuk fel. A tanuló halmaz tehát a korpusz 200 ezer szóból álló szegmense volt, amely a teljes mondatszintaxisra nézve tartalmaz bejelöléseket, és amelyet a megfelelő tulajdonnévi osztályok kódjaival is elláttunk. Modellünkben a szintaktikai jegyekből csak a szófaji kódokat, esetragokat használtuk fel, a Humor elemzőre, egy végződéstippelő program eredményeire (ismeretlen szavakon), valamint POS taggerre támaszkodva. A 200156 szövegszóból 25382 képezi tulajdonnév részét, ezek a következő megoszlást mutatják a különböző kategóriák között:

1541 db helységnév, 19982 db szervezet, 2124 db személynév, 1735 db egyéb tulajdonnév

3 A tanuló modell

A modellhez minden szóhoz magára a szóra és a környezetére vonatkozó, numerikusan kódolható információkat gyűjtöttünk le (ezek egy részénél a [3] cikkben ismer-

tetett modellt vettük alapul), ezek szerepeltek az osztályozásnál az adott elem attribútumaiként. Magát a szóalakot nem használtuk fel, mint attribútumot.

Az általunk használt jellemzők rendre a következők:

- Szófaji kód (magára a szóra, és +/-4 szavas környezetére)
- Esetrag
- Kezdőbetű típusa (magára a szóra és +/-4 szavas környezetére)
- Tartalmaz-e számjegyet (a szó belsejében)
- Tartalmaz-e nagybetűt (a szó belsejében)
- Tartalmaz-e írásjelet (a szó belsejében)
- Modateleji szó-e
- Idézőjelek közt szerepel-e a mondatban
- Szóhossz
- Arab vagy római szám-e
- A Szószablya [4] gyakorisági szótárban a kisbetűs és összes előfordulás hányadosa
- A Szószablya gyakorisági szótárban a mondatközi nagybetűs és összes nagybetűs előfordulás hányadosa
- Szerepel-e a szó valamely szótárunkban (településnevek, országnevek, keresztnevek, cégnevek utótagjait, földrajzi nevek utótagjait, valamint tulajdon-névben gyakori kisbetűs szavakat tartalmazó szótárakat használtunk)

A fenti jellemzők felhasználásával minden szóalakhoz 37 különböző attribútum tartozott, valamint a megfelelő osztály kódja. Az osztályozásra C4.5 döntésifa-tanuló algoritmust alkalmaztunk, melyet a WEKA programcsomagból [8] használtunk fel.

A módszer értékeléséhez egy meglehetősen jól működő naiv algoritmust használtunk, ami a következőképpen működött: *Minden nagybetűs szóra, amely tulajdonnév, a <szervezet> osztályt rendelte.* Ez az egyszerű eljárás a validációs halmazon 71.9%-os precision, 69.8% recall értékeket ért el (70.8% F mérték). A jó eredmény annak köszönhető, hogy hozzáadtuk azt az információt, hogy mi tulajdonnév (tehát a mondat eleji nagybetűs szavak nem okoztak tévesztést), ami sok információt adott a modellhez, másrészt az alapul vett szövegekben a <szervezet> osztály dominálja a másik 3 kategóriát, ami kedvez a valószínűségi taggernek.

4 Kísérletek

A tanuló modell kiértékelésére 3 különböző kísérletet végeztünk, melyek eredményei láthatóak az 1. táblázatban. A kísérletek során vizsgáltuk a tanult modell pontosságát, és a döntési fa méretét a tanuló halmaz nagyságának függvényében. Az adathalmaz 96 db XML-fájlból állt, melyből validációs célokra a korpuszból véletlenszerűen válogatott 9 db fájlt használtunk (A teszthalmaz 5, a tanuló halmaz 82 fájlból állt).

Az első kísérletben a rendelkezésre álló, megközelítőleg kétszázezer példából csak annak kis részét (mintegy 9-10%-ot) használtuk fel a tanuláshoz, melyen 10-szeres keresztvalidációt végeztünk, a halmaz kis mérete miatt. A kapott döntési fa 155 csúcsot és 78 levelet tartalmazott, a validációs halmazon 82.8% precision, 86.7% recall, 84.7% F mérték[2] eredményeket ért el. Ezek az eredmények meghaladják az összeha-

[2]Precision: Helyesen osztályozott tulajdonnevek és az összes tulajdonnévnek osztályozott példa hányadosa

sonlításhoz használt naiv algoritmus eredményeit, ami elsősorban a <szervezet> osztály sikeres felismerésének köszönhető, a másik 3 kategórián a pontosság gyengébb.

A második kísérletben a korpuszból előállt példák felét használtuk fel a tanuláshoz, ezúttal keresztvalidáció nélkül (elegendően nagy példahalmaz), egy rögzített tesztelhalmazt használva a modell értékelésére. Ezúttal a döntési fa 433 csúcsból, 217 levélből állt, és 85.4% precision, 86.9% recall, 86.1% F mérték pontosságot ért el. A javuló eredményeket döntően 3 kategórián mérhető pontosságnövekedés okozta, míg meglepő módona leggyengébb pontosságot adó osztály felismerése romlott (egészen 1%-ra, azaz gyakorlatilag nem ismert fel pontosan elemet a fa ebbe a kategóriába, az <egyéb> tulajdonnevek döntően <szervezet> cimkét kaptak).

Harmadik esetben a teljes rendelkezésre álló adathalmazt használtuk. Az eredményül kapott döntési fa 839 csúcsból, 420 levélpontból állt, és 91% precision, 89.7% recall, 90.3% F mérték pontosságot mutatott. Mivel ezek voltak az eddigi legjobb eredmények, 10 különböző futtatást végeztünk (más-más tanuló, teszt, validációs halmazokon). Az eredmények átlagban nem sokkal maradtak el az előzőleg használt futtatás eredményeitől (89.6%-os átlagos pontosság, 1.86% szórással).

1. táblázat: A különböző vizsgálatok eredményei osztályonként, és átlagban:

	1. Kísérlet	2. Kísérlet	3. Kísérlet	3. Kísérlet (10 futás átl.)
	F measure			
Helységnév	57.5%	64.2%	68.5%	74.2%
Szervezet	90.2%	92.3%	94.5%	93.8%
Személynév	65.3%	67.6%	74.4%	76.5%
Egyéb	11.0%	1.0%	54.0%	59.5%
Átlagos pontosság	**84,7%**	**86,1%**	**90.3%**	**89.6%**
Javulás a baseline-hoz képest	**19,6%**	**21,6%**	**27.5%**	**26.6%**

5 Konklúzió, további lehetőségek

Az eredmények alapján elmondható, hogy a tulajdonnevek felismerésének statisztikai módszerekkel való megközelítése magyar szövegekben is eredményes lehet. Természetesen a tanuló modellt számos további attribútummal lehetne finomítani (pl. az adott szó kapott-e már tulajdonnév címkét az aktuális szövegben, és ha igen, milyet), valamint a döntési fák által előállított osztályozások alacsonyabb pontossággal rendelkező osztályait további vizsgálatoknak is alá lehetne vetni. További feladat a rendszer felkészítése az egymás melletti, azonos típusú tulajdonnevek szeparálására (szakértői szabályokkal, vagy a modellbe építve), valamint tervezzük az eredmények összeveté-

Recall: Helyesen osztályozott tulajdonnevek és az összes tulajdonnév token hányadosa

F mérték: A Precision és Recall értékek harmonikus közepe

sét a HumorESK program tulajdonnév felismerő funkciójával, a két rendszer kombinálhatóságának vizsgálatát.

Bibliográfia

1. Conference on Computational Natural Language Learning (CoNLL-2003, 2002): Language-Independent Named Entity Recognition. http://cnts.uia.ac.be/signll/conll.html (2003)
2. Csendes Dóra, Csirik János, Gyimóthy Tibor: The Szeged Corpus: A POS Tagged and syntactically Annotated Hungarian Natural Language Corpus. In Sojka et al., 41–47 (2004)
3. Curran, James R., Clark, Stephen: Language Independent NER Using a Maximum Entropy Tagger. Proceedings of CoNLL-2003, 164–167, Edmonton, Canada (2003)
4. Halácsy P., Kornai A., Németh L., Rung A., Szakadát I., Trón V.: A szószablya projekt – www.szoszablya.hu. MSZNY 2003, 298–299, Szeged, Magyarország (2003)
5. Miháczi András, Németh László, Rácz Miklós: Magyar szövegek természetes nyelvi előfeldolgozása. MSZNY 2003, 38–44, Szeged, Magyarország (2003)
6. Prószéky Gábor: Syntax As Meta-Morphology. Proceedings of COLING-96, Vol.2, 1123–1126. Copenhagen, Denmark (1996)
7. Quinlan, J. R.: C4.5: Programs for machine learning, Morgan Kaufmann. (1993)
8. Witten, I. H., Frank, E.: Data Mining: Practical machine learning tools with Java implementations, Morgan Kaufmann, San Francisco, USA (2000)

Többszavas kifejezések számítógépes kezelése

Oravecz Csaba, Varasdi Károly és Nagy Viktor

MTA Nyelvtudományi Intézet, Budapest
{oravecz,varasdi,nagyv}@nytud.hu

Kivonat Azok a több szóból álló kifejezések, melyek tulajdonságainak egy része nem következik a nyelvtan sztenderd szabályaiból, a nyelvi elemzés valamilyen szintjén egy egységként jelennek meg. A számítógépes nyelvfeldolgozás során ezeket az egységeket képesnek kell lennünk azonosítani, és hozzájuk rendelni a produktív szabályokból nem következő jellemzőiket. Ez a feladat a hazai számítógépes nyelvészetben súlyához képest eddig kevés figyelmet kapott, ezért a dolgozat egy lehetséges azonosító, kinyerő módszer értékelése mellett a többszavas kifejezések kezelésének általános problémáit is tárgyalja.
Kulcsszavak: többszavas kifejezések, szinonimitás, idióma, lexikai idioszinkrázia, helyettesíthetőség

1. Bevezető

A számítógépes nyelvfeldolgozás szinte minden területén, különösen pedig a minél részletesebb („mély") nyelvi elemzést igénylő feladatokban a többszavas kifejezések kezelése az egyik legnagyobb kihívást jelentő probléma. Egészen a legutóbbi időkig még az ilyen irányú nemzetközi kutatást sem tartották elegendő mértékűnek (Sag et al., 2002), magyar nyelven pedig egy viszonylag jól körülírható típussal foglalkozó kezdeti próbálkozások kivételével[1] a probléma kutatása éppen, hogy elkezdődött (Kis et al., 2004; Kis és Ugray, 2003). Fontosnak tartjuk ezért azt, hogy a magyar nyelvvel foglalkozó számítógépes nyelvfeldolgozás szempontjait tekintve elsődlegesnek néhány alapvető kérdést megpróbáljunk összefoglalni és tisztázni. Kiindulásként kísérletet teszünk annak meghatározására, hogy a magyar nyelvben milyen szósorozatokat tekinthetünk *többszavas kifejezésnek* (TSZK), ezeknek milyen típusait érdemes megkülönböztetni, milyen problémát jelent a számítógépes kezelésük, s milyen eszközök és eljárások állhatnak rendelkezésünkre, ha ezt a problémát le szeretnénk küzdeni. Mint minden (kezdeti állapotban lévő, folyó) kutatásban, nagy mértékben támaszkodunk a külföldi szakirodalomra, de egyúttal szeretnénk azt is bemutatni, hogy az eddigi eredmények milyen módon hasznosíthatók, illetve milyen sajátos problémákkal kerülünk szembe a magyar nyelvre irányuló számítógépes nyelvészeti kutatásokban.

[1] A „nyílt tokenosztályok" kezelése (*named entity recognition*). Éppen ezért ezzel a típussal a jelen dolgozat csak az említés szintjén foglalkozik.

A dolgozat a következőképpen épül fel. A 2. rész rövid leírást ad a TSZK-król általában. A 2. rész a TSZK-k típusait tárgyalja, míg a 3. részben számítógépes kezelésük problémáit mutatjuk be. A 4. rész lehetséges nyelvi diagnosztikai eszközöket ír le, majd az 5. rész egy kinyerő/azonosító eljárást értékel, és megmutatja, hogy az alapjául szolgáló hipotézis a magyar nyelvű korpuszban nem érhető tetten. Rövid összefoglalás zárja a dolgozatot a 6. részben.

2. Mi az a TSZK?

Az, hogy valójában mit tekintünk többszavas kifejezésnek, vizsgálható a nyelvleírás, nyelvelmélet (Chomsky, 1980; Pulman, 1993; Nunberg et al., 1994) és a számítógépes nyelvfeldolgozás, illetve ennek részterületei szempontjából is.[2] Mivel még az előbbi is jellemzően hol szűkebb, hol tágabb értelemben vizsgál több szóból álló kifejezéseket, meglehetősen reménytelen próbálkozás egy olyan egzakt, részletes definíciót adni, ami aztán a számítógépes nyelvészetben is minden területen egyformán jól alkalmazható lenne. Legáltalánosabban Sag et al. (2002) és Calzolari et al. (2002) nyomán a következőt lehetne definíciónak tekinteni: számítógépes nyelvészetben többszavas kifejezésnek (TSZK) nevezünk egy olyan idioszinkratikus értelemmel rendelkező szósort[3], ami a nyelvi elemzés valamilyen szintjén egy egységként jelenik meg. Ez aztán az alábbiak közül egy vagy több tulajdonsággal rendelkezhet:

- nem teljes kompozícionalitás, nem teljesen megjósolható jelentés. A kifejezés vagy megjósolhatatlan módon kétértelmű: a szószerinti jelentése mellett van egy átvitt értelme is, ami nem tudható be sem lexikai, sem szintaktikai kétértelműségnek (*húzza a lóbőrt, kiborítja a bilit*), vagy van valamilyen hozzáadott jelentés, ami nem megjósolható az elemek jelentéséből (*kaparós sorsjegy*).
- többé kevésbé rögzített forma. A kifejezés egy folytonos skálán helyezkedik el, melynek egyik végén az egyáltalán nem változtatható kifejezések állnak (*ad hoc*), majd azok, ahol szerepel legalább egy olyan elem, ami nem létezik más környezetben (*hiszi a piszi, kidobja a taccsot*), vagy elemcsoport, ami ilyen (*a füle botját se mozdítja*), másikon a egyes elemek helyettesítését nem toleráló TSZK-k állnak (**vakarós sorsjegy*[4]).
- megsérti a szintaxis szabályait (*Ezt nevezem!* vs. *ezt vminek nevezem*).

A TSZK-k gyakorlatilag a nyelvtan és a lexikon közötti területen helyezkednek el. Ezekből a tulajdonságokból, valamint abból a tényből, hogy a TSZK-k

[2] A bevezetőben említett előzménynélküliség bizonyos mértékig a magyar általános nyelvészeti szakirodalmat tekintve is igaz. Lásd pl. Forgách (1998), aki „mostohán kezeltnek" tekinti a területet.

[3] A *szósor* inherens tulajdonsága, hogy szóhatárt, ami általában szóköz karakter, tartalmaz.

[4] Megjegyzés: a következőkben a '*' jel nem feltétlenül szintaktikai rosszulformáltságot jelöl, hanem azt, hogy egy idiomatikus kifejezés egy adott változatban már nem rendelkezik idiomatikus olvasattal.

száma igen nagy, illetve a fenti jellemzőket tekintve egy a teljesen kompozícionális, produktív szókapcsolatoktól a teljesen idiomatikus és rögzített alakokig tartó kontinuum mentén helyezkednek el, több probléma következik (mind elméleti mind) számítógépes szempontból. Egyrészt nehéz egzakt, általános kategóriarendszer alapján adott típusokba sorolni, illetve egyáltalán azonosítani őket. Az, hogy adott esetben szükség van-e arra, hogy adott kombinációt TSZK-nak tekintsünk, függ a feldolgozás módjától és mélységétől (pl. egy szintaktikailag reguláris TSZK nem releváns, ha a rendszer kimenete csupán szintaktikai elemzés). Másrészt, mivel ezek a szókapcsolatok olyan jellemzőkkel rendelkeznek, melyek nem következnek a nyelvtan sztenderd szabályaiból, az adott TSZK-khoz hozzá kell tudni rendelni ezeket a jellemzőket. Ehhez a nyelvfeldolgozás során természetesen képesnek kell lennünk azonosítani a kifejezéseket, illetőleg a lexikonban történő eltárolásukhoz alkalmas reprezentációs formalizmust kell felhasználnunk. A következő részben javasolt osztályozás után mind a két problémát tárgyaljuk az alábbiakban.

2.1. TSZK típusok — egy lehetséges osztályozás

A TSZK-k osztályozásában sincs sokkal nagyobb konszenzus és általánosan elfogadott sztenderd, mint az azonosításukban. Mindesetre két alapvető szempontot meg tudunk különböztetni, és ennek alapján kétféle kategorizálást is végezhetünk: egy operacionálist és egy grammatikai tulajdonságokon alapulót. Természetesen mindegyik osztályozás jár bizonyos következményekkel a számítógépes feldolgozásban használható módszerek és lehetőségek tekintetében.

Az előbbi szempont szerint triviális módon, a TSZK-ban szereplő elemek egymás után következésének függvényében különböztethetünk meg kétféle típust, **folytonos** (*bakot lőtt*) és **megszakított** (*megkongatja a vészharangot* vs. *a vészharang, amit a tudósok megkongattak*) kifejezéseket. Ugyan a TSZK-ban szereplő elemek sorrendje sokszor kötött (van egy *kanonikus sorrendje* a szavaknak), a magyar nyelvben igen gyakran találhatunk sorrendi variánsokat, akár nem a TSZK-hoz tartozó elemek beékelődésével együtt. Ezzel a problémával mindenképpen szembekerülünk a számítógépes elemzés/feldolgozás során.

A grammatikai jellemzők alapján Bauer (1983), Nunberg et al. (1994) és Sag et al. (2002) terminológiáját és osztályozását felhasználva az alábbi típusokat különíthetjük el:

- **intézményesült kifejezések**. Szintaktikailag és szemantikailag kompozícionálisak, statisztikailag idioszinkratikusak[5], vagyis az adott jelentés hordozóiként sokkal gyakrabban fordulnak elő, mint más hasonló kompozícionális kombinációk, illetve akár blokkolhatják is a jelentés alternatív realizációját[6].
- **funkcióigés kifejezés**. A teljesen lexikalizált alakoktól (*részt vesz, pofon vág*) a „terpeszkedő kifejezésekig" (*javaslatot tesz*).

[5] Talán azt mondhatjuk, hogy a „sztochasztikus kompozícionalitás" feltételének nem felelnek meg.

[6] Pl. *közlekedési lámpa* vs. **forgalmi lámpa*, *közlekedési *világítótest*. Az ilyen módon blokkolt szókapcsolat egyébként *anti-kollokáció* néven ismert (Pearce, 2001).

- **ige + partikula szerkezet.** Ide tartoznak az igekötős igék elváló alakjai (akár ragos névszós igekötővel: *létrehoz, észrevesz*).
- **féligáttetsző idiómák.** Összetevői jól megfeleltethetők az idiomatikus jelentésükben szereplő összetevőknek. Ezt onnan láthatjuk, hogy pl. *bakot lő* idióma *bak* részkifejezése szisztematikusan módosítható olyan kifejezésekkel, amikkel a *hibákat* lehet jellemezni: *nagy/komoly/végzetes/elképesztő bakot lőtt* ill. *nagy/komoly/végzetes/elképesztő hibát követett el.* Ez a szisztematikusság hiányzik a következő típusban szereplő kifejezésekből.
- **homályos idiómák.** Szemantikai transzparencia hiányában ezeknek a kifejezéseknek az összetevői nem vethetők alá módosításoknak az idiomatikus jelentés elveszése nélkül. Ennek következtében ez a típus gyakorlatilag csak inflexiós variánst tűr meg (*felveszi a kesztyűt* vs. *felvette a *politikai kesztyűt*).
- **többszavas tulajdonnevek.**
- **összetett szavak.** Pontosabban a helyesírási szabályok miatt szóközt tartalmazó összetett szavak (*nagy néha, nyitva tartás*).
- **rögzített kifejezések.** Semmilyen módosítást nem engednek meg, csak egyféle alakban léteznek (*így vagy úgy, egytől egyig*).

Ez az osztályozás, amint látni fogjuk, sajnos nem képezhető le közvetlenül a feldolgozó eljárások különböző típusaira.[7]

3. A TSZK-k feldolgozásának problémái

3.1. TSZK-k azonosítása, korpuszból történő kinyerése

A TSZK-k számítógépes elemzésének legelfogadottab módja lexikonban való tárolásuk, és abból történő kiolvasásuk[8]. Ebből következően a legfontosabb feladat korpuszból történő automatikus kigyűjtésük és osztályozásuk, amelynek érdekében ritkán tisztán szimbolikus, előre meghatározott szabályokon alapú (Bourigault, 1996), sztohasztikus (Church és Hanks, 1990; Dunning, 1993; Shimohata et al., 1997), illetve leginkább vegyes rendszereket (Daille, 1996; Heid, 1999) használnak. Az előbbiek alapvető problémája a nyelvspecifikusság, ám a klasszikus statisztikai alapú rendszereknek is van egy komoly hátránya. Ezek ugyanis a

[7] Érdemes itt egy terminológiai kitérőt is tenni és megjegyezni, hogy a gyakran használt *kollokáció* kifejezés alatt sokak leginkább az itt *intézményesült kifejezésnek* nevezett típust értik (McKeown és Radev, 2000). Elterjedt azonban egy jóval átfogóbb értelmezés is, amely szerint kollokáció minden szignifikánsan gyakran előforduló szókapcsolat, vagyis az összes fenti TSZK (Manning és Schütze, 1999), valamint akár az egyéb (nem nyelvi) okok miatt gyakran együtt előforduló teljesen produktív és kompozícionális szókapcsolat is (pl. ⟨*apróhirdetés, felad*⟩) (Sag et al., 2002).

A Nunberg et al. (1994) által bevezetett szemantikai dekomponálhatóság szempontjából szokás egyébként az itt féligáttetszőnek ill. homályosnak nevezett idiómákat dekomponálható ill. nemdekomponálható idiómáknak is nevezni.

[8] Bár van példa futási időben működő, általában szabály alapú TSZK elemző rendszerre is (Oflazer et al., 2004).

TSZK-knak azt az egyik fontos tulajdonságát használják ki, hogy elemeik általában gyakrabban fordulnak elő együtt, mint egyéb önkényes szókombinációk, és ennek az együtt előfordulásnak az erősségét számszerűsítik valamilyen *asszociációs mérték* (AM) segítségével. Csupán ennek a mértéknek a használata azonban rendkívül zajos eredményre vezet, különösen, ha a gyakorisági alapú mutató megbízhatóságának növelése és az osztályozás finomítása érdekében minél nagyobb mennyiségű korpuszt használunk. Ez a probléma szabad szórend esetén még élesebben jelentkezik (Kaalep és Muischnek, 2002).

Egy lehetséges megoldás egyrészt a feladat leszűkítése (vagyis a feladat nem általában TSZK-k keresése, hanem valamelyik jól meghatározott összetevőkből álló és szerkezetű részosztályé[9]), ehhez azonban szükséges a felhasznált korpusz minél részletesebb nyelvi annotációja, és ennek az annotációnak a legalaposabb kihasználása. Kérdés azonban, hogy honnan származik és egyáltalán rendelkezésre áll-e ez az annotáció?

Természetesen az ideális eset emberi tudás felhasználását nem igénylő nemfelügyelt tanuló eljárás(ok) alkalmazása lenne, amely nyers korpuszból képes lenne adott típusú TSZK azonosítására, és ugyan történtek kísérletek ebben az irányban (Schone és Jurafsky, 2001), biztató eredményt nemigen hoztak. Jelenleg tehát kénytelenek vagyunk opportunista megközelítést választani: mivel nincs egyedül célravezető, minden típusú TSZK kinyerési feladatra alkalmas módszer (Krenn és Evert, 2001), szűkítsük a szóba jöhető jelöltek körét egy jól meghatározott típusra, és használjunk fel minden lehetséges nyelvi erőforrást az adott típushoz igazított kinyerési módszerhez.

Ezt a megközelítést magyar nyelvre több szempont is indokolja. Ha a szórendi változatosságot legalább bizonyos mértékig figyelembe vevő nem minimális számú (2-3) szóból álló szókapcsolatok tetszőleges sorozataiból képezünk jelöltlistát, a nagy számú, nagy variábilitással bíró elem miatt nagyobb korpusz használata esetén implementációs, hatékonysági problémákba ütközhetünk, ezért célravezető a jelöltlista típusos szűkítése. Ehhez persze a korpusz minimális (POS) annotációjára legalább szükség van. A szórendi variábilitás következtében pedig a pozíciós szókapcsolatok helyett relációs szókapcsolatokon célszerű jelöltlistát definiálni, ehhez viszont függőségi viszonyokat is tartalmazó annotáció kell.[10]

A fentiek alapján az alábbi főbb lépéseket látjuk fontosnak egy magyar szövegen működő TSZK kinyerő módszer létrehozásában:

- az azonosítani kívánt TSZK altípus illetve jelenség pontos meghatározása
- nyelvtani jellemzők, viselkedés részletes feltárása
- ennek alapján specifikus eljárás kidolgozása és alkalmazása.

Ezeket a lépéseket követő prototípus eljárást mutatunk be az 5. részben, előtte azonban röviden érintjük a kinyert és azonosított TSZK-k lexikonbeli reprezentációjának kérdését, majd összefoglaljuk azokat a nyelvi jelenségeket, amelyek diagnosztikai eszközként szolgálhatnak a kinyerési módszerek számára.

[9] Kis et al. (2004) ezt *típusos kollokációnak* nevezi.

[10] Amíg ez nem áll rendelkezésre, a POS annotáción működő reguláris szabályokkal közelíthető.

3.2. Reprezentáció

Ha rendelkezésünkre áll a TSZK-k listája és releváns tulajdonságaikat is meghatároztuk, a nyelvfeldolgozó rendszerekben történő hatékony hasznosíthatóságuk érdekében egyértelmű és gépileg kezelhető módon kell tárolnunk őket. A legegyszerűbb, változtathatatlan lexikai egységként (*word_with_spaces*), „listéma"-ként (Sciullo és Williams, 1987)való egyszerű felsorolás csak a *rögzített kifejezés* típusú TSZK-k kezelésére alkalmas. Egyéb esetben az összes variáns felsorolása kivihetetlen. A sztenderd nyelvtani szabályok által történő kezelés pedig a túlgenerálás és az idiomatikus jelentés származtatásának problémájával néz szembe.

Mivel jelenleg nincs kidolgozott, nyelvészetileg jól megalapozott, általánosan elfogadott számítógépes nyelvtani rendszer magyarban, jól kezelhető és a releváns tulajdonságokat egységesen leíró reprezentációs formalizmust pedig nehéz ettől teljesen függetlenül kidolgozni (Villavicencio et al., 2004), ismét egy opportunista következtetésre vagyunk kénytelenek jutni. A jelen helyzetben legfeljebb konkrét alkalmazásokhoz lehet specifikus TSZK (és nehézkesen hordozható) erőforrásokat fejleszteni.

4. TSZK-k nyelvi diagnosztikai eszközei

A következőkben néhány lehetséges eljárást vázolunk a TSZK-k automatikus kivonatolására. Ezek eltérő mértékben előfeldolgozott korpuszt igényelnek, ezért kívül eshetnek jelenlegi lehetőségeink határain — fontosságukat az adja, hogy kijelölnek néhány követhető jövőbeli kutatási irányt.

Elméleti szempontból a TSZK-k a „normálishoz képest" csökkent variabilitással bíró kifejezések. Bár ez a megfogalmazás a normalitás homályossága miatt maga is meglehetősen homályos, mégis talán úgy pontosítható, hogy a TSZK-k bizonyos, a szintaxis elvei által jósolt változatokkal nem rendelkeznek. Ez a hiány nem vezethető le sem a TSZK-ban előforduló kifejezések kategóriája, sem pedig a TSZK szintaktikai szerkezete alapján. Más szóval, az idiómák meglehetősen gyengén vethetők alá különböző szintaktikai transzformációknak idiomatikus jelentésük elvesztése nélkül. Ebben azonban az egyes idiómák között jelentős eltéréseket találunk. Az alábbiakban néhány példát mutatunk arra, ahol ez a jelenség tetten érhető.[11]

Eldöntendő érdés: *A bolondját járatod velem?* DE: **Hiszi a piszi?*

Kiegészítendő-kérdés: *Miféle vészharangot kongattak meg a tudósok?* DE: **Mit járattál Jánossal?*, **Miféle csatabárdot ástak ki Mariék?*, **Melyik kesztyűt vette fel Béla?*

Igeidő: *minden követ meg fog mozgatni, minden követ megmozgatott*, DE: **hinni fogja a piszi, *hitte a piszi.*

Progresszív (aspektus): *Amikor beléptem, János éppen húzta a lóbőrt*, DE: **Amikor beléptem, Jánosék éppen ásták ki a csatabárdot.* Ennek a magyarázata az lehet, hogy a *kiássa a csatabárdot* idiomatikus jelentése ('nyílt ellenségeskedésbe kezd') nem jól progresszivizálható (mivel egyfajta achievement), ezért az

[11] A '*', mint már említettük, itt az idiomatikus jelentés hiányát jelzi.

idióma formai progresszivizálása csak a szószerinti értelmet adhatja (hogy ui. Jánosék nagyban egy konkrét indián harci eszköz földből való kiemelésén dolgoztak).

Topikalizáció: *a rizsát, azt tudja nyomni, bakot, azt nem lőtt, de majdnem*, DE: **János a törülközőt bedobta, *a csatabárdot, azt kiásták/?a csatabárdot kiásták, *a kulcsot beadta a beteg, *a hajó elment, *a kesztyűt fel szokta venni.*

Fókusz: *'elásták a csatabárdot (és nem 'kiásták azt)*, DE: **a 'csatabárdot ásták el (és nem mást).*

Belső módosítás: *szép nagy bakot lőttél ezzel*, DE: **elment az utolsó hajó* (≠ 'elmulasztottad az utolsó lehetőséget').

Mellékmondatos módosíthatóság: *A vészharang, amit a tudósok megkongattak végül a washingtoni bürokratákat is felébresztette.* DE: **a bak, amit lőtt, végül az állásába került.*

Melléknévi igenévképzés: *a Jánossal a bolondját járató fiú, a csatabárdot újra kiásó ellenfelek, a minden követ megmozgató alperes*, DE: **az elmenő hajó* (≠ 'az eltűnő lehetőség'), *?a törülközőt bedobó ügyfél, *a kulcsot beadó beteg.*

Nominalizáció: általában nem lehetséges — **a jég (köztük történő) megtörése, *Jánosnak a Mari általi bolondját járatása, *a hajó elmenése, *a bak (le)lövése, ?a csatabárd kiásása, *minden kő megmozgatása*, bár pl. *a bili kiborulása nagy felbolydulást okozott* esetleg elfogadható idiomatikus jelentésben is.

A fenti tulajdonságok azonban — bár elméletileg relevánsak — közvetlenül nem használhatók fel jelenlegi céljaink eléréséhez, hiszen azt jelölik ki, ami nem létezik, míg a korpusz annak a tárháza, ami valóban aktualizálódott. Ezért az alábbiakban a TSZK-k olyan általános jellemzőire koncentrálunk, amelyek segítségével a korpuszból ténylegesen kinyerhetőkké válnak az ilyen kifejezések.

4.1. Tematikus inkongruencia

A TSZK-k egyik legalapvetőbb jellegzetessége (formai kötöttségük mellett), hogy jelentésük nem (teljesen) kompozícionális. Ugyanakkor azonban a TSZK-k igen nagy részéhez tartozik közvetlen, kompozícionális jelentés is. Ennek a ténynek a kommunikáció során is komoly jelentősége van, amelyet a kommunikáló partnerek kénytelenek figyelembe venni: *egy TSZK csak akkor használható idiomatikus jelentésében, ha kellő tematikus inkongruencia áll fenn a diskurzus tematikája és a TSZK kompozícionális jelentése között.* Az idiomatikus jelentés a szövegtematikába pragmatikai vagy egyéb okonál fogva be nem illeszthető kompozícionális jelentés kizárása után válik relevánssá a hallgató számára mint olyan „másodrendű jelentés," amelynek invokálásával képes elkerülni a kommunikáció megakadását (*Principle of Charity*, ld. Davidson (2001)). Ennek a körülménynek a számítógépes használatba vétele a szöveg „szemantikai súlypontjának" meghatározását igényli, ám ez elvileg és gyakorlatilag is lehetséges a jelenleg is használatban lévő vektoralapú szövegosztályozó eljárások segítségével. Az inkongruens, azaz nagy valószínűséggel idiomatikus jelentésű kifejezések hatása abban nyilvánulhat meg, hogy a szöveghez rendelt vektor a kifejezés hozzáadása után olyan mértékben megváltozik, amely egyébként a diszkurzus befejezését és egy új topik megnyitását jellemezné.

Bár a tematikus inkongruenciára építő eljárások teljes általánosságukban a diskurzustopik azonosítására alkalmas eszközöket igényelnek, az alábbi pontban tárgyalandó sajátos eset már szerényebb keretek között is detektálható lehet.

4.2. Szemantikai inkongruencia

A TKSZ-ek esetében sokszor találkozunk szemantikai furcsaságokkal: pl. *a szőnyeg alá söpri a problémát* esetében látszólag egy fizikai cselekvést (*seprés*) alkalmazunk egy absztrakt entitásra (*probléma*), ami első látásra kategóriahiba. Ehhez hasonló még: *húzza az időt, bedob egy új témát, kikerüli a választ.* Ez az inkongruencia már akkor is észlelhető, ha a korpusz szemantikailag legalábbis minimálisan annotálva van. Az ilyen szemantikailag anomális idiómák egyébként szintaktikailag úgy tűnik nagyobb variációs szabadságot is engednek meg: topikalizálhatók és belsőleg módosíthatók (*Az ilyen nem életbevágó problémát általában megpróbálják a szőnyeg alá söpörni az illetékesek, hacsak valaki meg nem akadályozza őket ebben*), továbbbá nominalizálhatóak is (*A probléma szőnyeg alá söprése nem jelent hosszú távú megoldást*). Ez valószínűleg azzal függ össze, hogy ez a fajta nyilvánvaló szemantikai anomália önmagában is elegendő az idiomatikus jelentés detektálásához minden környezetben, ezért nincs szükség további megszorítások kirovására.

4.3. Lexikai idioszinkrázia

Láttuk, hogy a TSZK-k alkatrészei sokkal kisebb variálhatósággal bírnak, mint a a teljesen produktív kifejezéseké. Ez többek között abban is megnyilvánul, hogy a TSZK részei többnyire nem cserélhetők fel (közel) szinonim kifejezésekkel az idiomatikus jelentés elveszése nélkül (aminek következtében — ha történetesen csak az idiomatikus jelentés létezett — a kifejezés értelmetlenné is válhat). Pl.: *a* ^{OK}*bolondját/*hülyéjét/ *gyengeelméjűjét/*retardáltját járatja vkivel*, ill. *játssza az* ^{OK}*eszét/*értelmét/ *intellektusát.* Ezt a jelenséget nevezhetjük **lexikai idioszinkráziának** (a szemantikai ekvivalensek közül is csak a „megfelelő" szó illeszthető be). Feltételezhetjük, hogy a teljesen produktív szókapcsolatok jobban tűrik ezt a fajta behelyettesítést, így ennek vizsgálatával lehetőség nyílhat TSZK-k azonosítására, illetve a TSZK-k teljesen produktív szókapcsolatoktól történő elkülönítésére.

A jelen vizsgálatban a TSZK-knak ezt a vonását kíséreltük meg tesztelni egy gépi szinonímaszótár felhasználásával.

5. TSZK-k és produktív szókombinációk elválasztása

A szinonimahalmazokon belül a helyettesíthetőség mértéke gyakorlatilag bármilyen AM segítségével kifejezhető, hiszen a csupán a hasonló jelentésű elemek előfordulására vonatkozó vizsgálattal nem teszünk mást, mint az AM számára az eseményteret leszűkítjük, ezáltal a mérték pontosságát illetve zajmentességét

próbáljuk növelni.[12] Kézenfekvő választás mint AM az a *kölcsönös információ* (KI), melyet valamilyen formában a hasonló, szemantikai alapú behelyettesíthetőséget vizsgáló eljárásokban többen alkalmaztak. Lin (1999) függőségileg elemzett korpuszból kinyert ⟨*fej,relációtípus,módosító*⟩ hármasokat vizsgált a kompozícionalitás szempontjából, míg McCarthy et al. (2003) frazális igék kompozícionalitásának erősségére kapott mértéket vetette össze emberi megítéléssel, mindegyikük elég visszafogott eredménnyel. Ez is jelzi azt, hogy a helyettesíthetőségen alapuló tesztek nem nagyon alkalmasak a kompozícionalitás mértékének meghatározására (Baldwin et al., 2003), úgyhogy célszerű inkább azt feltételezni, hogy csupán a produktív kombinációkat képesek elválasztani a TSZK-któl általában. Ezért az alábbiakban mi is ezzel a feladattal próbálkozunk.

Első lépésben AM-ként a Pearce (2002) által javasolt, a KI értékhez hasonló „standardizált eltérést" használjuk az alábbi módon. A szinonimahalmazok forrásaként a Magyar Szókincstár (Kiss, 2001) elektronikus változata szolgál.[13] Jelöljük $\mathcal{D}$-vel a szótári adatbázist, ebből rendelhetjük hozzá egy-egy szóhoz a hozzá tartozó szinonimahalmazokat:

$$\mathcal{D} = \{S_1, S_2, S_3 \ldots\} \tag{1}$$

Egy F fogalom egy lehetséges lexikális megvalósítása legyen K többszavas kifejezés, melyre: $K = \langle w_1 \ldots w_n \rangle$, mint elemi esemény. Az F fogalom összes lehetséges megvalósítása alkotja az eseményteret, mely a következőképpen definiálható a szinonimahalmazokon:

$$\Omega(K) = \{w'_{1,n} : w'_i \in S_i, 1 \leq i \leq n\} \tag{2}$$

ahol S_i a w_i szónak megfeleltethető szinonimahalmaz.

Ha feltesszük, hogy K teljesen produktív kifejezés, és elemeit egymástól függetlenül választjuk a megfelelő szinonimahalmazból, akkor K_i előfordulási valószínűségét a következőképpen közelíthetjük:

$$\hat{p}(K_i) = \prod_{i=1}^{n} p_i(w_i) \tag{3}$$

$p_i(w_i)$ annak valószínűsége, hogy az S_i szinonimahalmazból éppen w_i-t választjuk:

$$p_i(w_i) = \frac{f(w_i|S_i)}{\sum_{w \in S_i} f(w|S_i)} \tag{4}$$

Az így kapott értéket összevethetjük az adott kifejezés tényleges előfordulásával:

$$p(K_i) = \frac{f(K_i)}{\sum_{K \in \Omega(K)} f(K)} \tag{5}$$

[12] Nyilván a szinonimahalmazok megkonstruálásához szükséges erőforrással ennek „meg is fizetjük az árát".

[13] Gépi úton korpuszból származtatott tezaurusz is használható (pl. Lin (1998)), ennek magyarra történő megalkotása és a két módszer alapján történő összehasonlítás azonban egy későbbi kutatás tárgya lehet. Pearce (2002) szintén kész adatbázist, WordNet synseteket alkalmaz.

A két érték közötti különbség, illetve ennek z-transzformáltja jelzi, hogy az adott kifejezés mennyire „tűri" a helyettesíthetőséget:

$$z_i = \frac{d_i}{\sigma(d)}, \quad \text{ahol} \quad d_i = p(K_i) - \hat{p}(K_i) \tag{6}$$[14]

Ha magas z értéket kapunk, a kifejezés TSZK-nak lenne tekinthető.

5.1. Kiértékelés

A vizsgálathoz szükséges jelöltlista előállításához az MNSZ (Váradi, 2002) (POS egyértelműsített[15]) teljes anyagát használtuk (153 millió szó), amelyből 3-féle listát állítottunk elő:

1. szomszédos melléknév+főnév (L1)
2. egy mondaton belüli igék és tárgyesetű főnevek minden lehetséges ige+főnév kombinációja (L2)
3. egy mondaton belüli határozott ragozású ige + tárgyesetű főnév párok (L3)[16].

Mivel ezúttal a kifejezések morfológiai idioszinkráziáját nem kívántuk vizsgálni, lemmatizált alakokkal dolgoztunk, és csak az 5-nél gyakrabban előforduló kombinációkat vettük figyelembe. Az értékelés során a fenti z értéken alapuló modellre M_z-vel hivatkozunk.

A kapott eredményeket egy olyan viszonyító alapmodellel vetettük össze, amelyben KI-t, illetve ennek t próbából számított küszöbbel szűrt változatát (Church et al., 1994) használtuk a teljes eseménytéren[17] (M_{va}). Az összehasonlítást érdemes elvégezni azzal a modellel is, amelyben a jelöltek rangsorolását a kifejezés elemeihez tartozó szinonimahalmazokból képezhető leggyakrabban előforduló (f' számú) elempár és a második leggyakoribb (f'' számú) elempár előfordulásából számított egyszerű gyakorisági arány ($s = \frac{f'-f''}{f'}$) végzi (M_s).

Az értékelésben gyakorlati szempontok miatt a legelterjedtebb, *legjobb n-lista* módszert alkalmaztuk: az egyes modellek által rangsorolt jelöltlistákból kiválasztottuk az első n (itt $n = 250$) kifejezést, és megnéztük, milyen arányban tartalmaznak TSZK-nak tekinthető szókapcsolatot (*pontosság*). Ugyan sokan (pl. Evert és Krenn (2001)) számolnak a másik közkeletű mérőszámmal is (*fedés*), egyszerűen belátható, hogy az ilyen típusú kiértékelési feladatokban ez semmiféle további új információt nem ad a felhasznált modellek minőségével kapcsolatban a *pontossághoz* képest[18]. Emiatt az 1. táblázat csupán *pontosság* értékeket tartalmaz.

[14] $\sigma(d) = \sqrt{\frac{\sum_i (d_i - \mu)^2}{n}}$

[15] Az egyértelműsítés hibája kb. 3%, ez a listában elkerülhetetlen zajhoz vezet.

[16] Szintaktikai annotáció hiányában ezzel az egyszerű heurisztikával próbáltunk közelíteni valamiféle fej-argumentum viszonyt.

[17] Vagyis azt mérjük, hogy a kifejezés mennyire összetartozó, de nem a szinonimahalmazokból alkotható variánsokhoz, hanem az összes lehetséges jelöltkombinációhoz képest.

[18] A *legjobb n-lista* esetén ugyanis a *pontosság*: $p(n) = \frac{TP(n)}{n}$, ahol TP(n) az n elemet tartalmazó listában a helyes találatok száma (*true positive*). A fedés ugya-

1. táblázat. A modellek teljesítménye.

Lista	Jelöltek száma	Pontosság $p(n = 250)$		
	$f(K_i) > 5$	M_{va}	M_s	M_z
L1	191454	54.4%	15.2%	17.2%
L2	452981	29.2%	4.8%	19.2%
L3	100559	56.8%	9.2%	38.0%

Az alapmodell (M_{va}) hasonló eredményt ad, mint a szakirodalomban található *pontosság* értékek, és igazolja a szokásos elvárást is: minél megszorítottabb a jelöltlista, annál hatékonyabb a kinyerő eljárás. A szinonimahalmazokban való helyettesíthetőségen alapuló modellek kudarcának oka véleményünk szerint az, hogy a 4.3. részben megfogalmazott, a *lexikai idioszinkráziát* kihasználó, és sokak (Lin, 1999; Pearce, 2001; McCarthy et al., 2003; Bannard et al., 2003) által használt hipotézis ugyan lehet, hogy igaz, de ez a korpuszban nem érhető tetten: nincs olyan mértékű mérhető különbség a teljesen produktív kifejezések elemeinek helyettesíthetőségében egy TSZK-k elemeihez képest, amit egy azonosító módszer jól fel tudna használni. Nem állítjuk azonban, hogy az eféle módszer teljesen haszontalan; előfordulhat, hogy egy nagyon részletesen specifikált TSZK altípus kinyerésében mégis használható[19], mindazonáltal egy általában szokásos módon megszorított jelöltlista esetén nem hoz eredményt.

6. Összefoglalás és további feladatok

A dolgozatban megkíséreltünk áttekintést adni a többszavas kifejezések számítógépes kezelése során felmerülő kérdésekről, és megoldandó feladatokról. Megpróbáltuk összefoglalni azokat a nyelvi jelenségeket, amelyek segítségével azonosító, kinyerő eljárások építhetőek, és megmutattuk, hogy a hasonló jelentésű elemek helyettesíthetőségén alapuló eljárások nem állnak szilárd, nagy korpuszból nyert adatokkal alátámasztható alapon.

Természetes további feladatként adódik magyar nyelven olyan vizsgálatok elvégzése, amelyek további információt adnak kinyerési módszerek alkalmazhatóságáról és hatékonyságáról: több féle AM alkalmazása és összehasonlító kiértékelése, illetve adott TSZK-típus azonosításához a legalkalmasabb AM kiválasz-

nitt: $f(n) = \frac{TP(n)}{C}$, ahol C a teljes listában található TP-k száma, ami konstans. Ekkor $f(n) = \frac{p(n) \times n}{C}$, vagyis a *fedés* valójában a *pontosság* információmentes, modellfüggetlen transzformáltja. (Ha $n = |$*teljes jelöltlista*$|$, akkor természetesen $f(n) = \frac{\frac{C}{n} \times n}{C} = 1 \rightarrow 100\%$. A *fedés* legfeljebb annyiban lehet informatív, hogy az adott módszerhez hozzá lehet rendelni azt a legkisebb $n < C$ értéket, melyre $f(n) = 1$, vagyis azt a minimális listanagyságot, amelyben az összes TP már benne van. Ezt a *pontosság* függvényről nem lehet leolvasni.)

[19] Ennek vizsgálata további feladat.

tása és kidolgozása. Ez a munka, konkrétan például a morfológiai idioszinkrázia jelenségét kihasználó módszer fejlesztése, jelenleg is folyik.

Hivatkozások

Baldwin, Timothy, Bannard, Colin, Tanaka, Takaaki és Widdows, Dominic. An Empirical Model of Multiword Expression Decomposability. In: *Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, Sapporo, Japan. 2003, 89–96.

Bannard, Colin, Baldwin, Timothy és Lascarides, Alex. A Statistical Approach to the Semantics of Verb-Particles. In: *Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, Sapporo, Japan. 2003.

Bauer, Laurie. *English Word-Formation.* Cambridge University Press, Cambridge, England, 1983.

Bourigault, Didier. Lexter, a Natural Language Processing Tool for Terminology Extraction. In: *Proceedings of 7th EURALEX International Congress*, 1996.

Calzolari, Nicoletta, Fillmore, Charles J., Grishman, Ralph, Ide, Nancy, Lenci, Alessandro, MacLeod, Catherine és Zampolli, Antonio. Towards Best Practice for Multiword Expressions in Computational Lexicons. In: *Poceedings of the Third International Conference on Language Resources and Evaluation*, Las Palmas, Spain. 2002, 1934–40.

Chomsky, Noam. *Rules and Representations.* Columbia Univeristy Press, New York, 1980.

Church, Kenneth W. és Hanks, Patrick. Word Association Norms, Mutual Information and Lexicography. *Computational Linguistics*, 1990, 16(1):23–29.

Church, Kenneth Ward, Gale, William, Hanks, Patrick, Hindle, Donald és Moon, Rosamund. Lexical Substitutability. In: Atkins, B. T. S. és Zampolli, Antonio szerk. *Computational Approaches to the Lexicon.* Oxford University Press, 1994, 153–180.

Daille, Béatrice. Study and Implementation of Combined Techniques for Automatic Extraction of Terminology. In: Klavans, Judith és Resnik, Philip szerk. *The Balancing Act: Combining Symbolic and Statistical Approaches to Language.* The MIT Press, Cambridge, Massachusetts, 1996, 49–66.

Davidson, Donald. Radical Interpretation. In: *Inquiries into Truth and Interpretation.* Clarendon Press, Oxford, 2001.

Dunning, Ted. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, 1993, 19(1):61–74.

Evert, Stefan és Krenn, Brigitte. Methods for the qualitative evaluation of lexical association measures. In: *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, Toulouse, France. 2001, 188–195.

Forgách, Tamás. Frazeológia és valencia. In: Büky, László és Maleczki, Márta szerk. *A mai magyar nyelv leírásának újabb módszerei*, III, 1998, 7–39.

Heid, Ulrich. Extracting Terminologically Relevant Collocations from German Technical Texts. In: Sandrini, P. szerk. *TKE99 Terminology and Knowledge*

Engineering. Proceedings Fifth International Congress on Terminology and Knowledge Engineering, Vienna. 1999, 241–255.

Kaalep, Heiki-Jaan és Muischnek, Kadri. Using the Text Corpus to Create a Comprehensive List of Phrasal Verbs. In: *Proceedings of the Third International Conference on Language Resources and Evaluation*, Las Palmas, Spain. 2002, 101–105.

Kis, Balázs és Ugray, Gábor. Új korpuszstatisztikai eszköztár kollokációkeresésre. In: *Magyar Számítógépes Konferencia*, Szeged. 2003, 131–136.

Kis, Balázs, Villada, Begoña, Bouma, Gosse, Ugray, Gábor, Bíró, Tamás, Pohl, Gábor és Nerbonne, John. A New Approach to the Corpus-based Statistical Investigation of Hungarian Multi-word Lexemes. In: *Poceedings of the Fourth International Conference on Language Resources and Evaluation*, Lisbon, Portugal. 2004, 1677–1680.

Kiss, Gábor szerk. *Magyar Szókincstár*. Tinta Könyvkiadó, Budapest, 2001.

Krenn, Brigitte és Evert, Stefan. Can we do better than frequency? A case study on extracting PP-verb collocations. In: *Proceedings of the ACL Workshop on Collocations*, Toulouse, France. 2001, 39–46.

Lin, Dekang. Automatic retrieval and clustering of similar words. In: *Proceedings of COLING/ACL-98*, Montreal. 1998, 768–774.

Lin, Dekang. Automatic identification of noncompositional phrases. In: *Proceedings of the 37th Annual Meeting of the ACL*, College Park, USA. 1999, 317–24.

Manning, Christopher és Schütze, Hinrich. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA, 1999.

McCarthy, Diana, Keller, Bill és Carroll, John. Detecting a Continuum of Compositionality in Phrasal Verbs. In: *Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, Sapporo, Japan. 2003.

McKeown, Kathleen R. és Radev, Dragomir R. Collocations. In: Dale, Robert, Moisl, Hermann és Somers, Harold szerk. *A Handbook of Natural Language Processing*. Marcel Dekker, 2000.

Nunberg, Geoffrey, Sag, Ivan A. és Wasow, Thomas. Idioms. *Language*, 1994, 70(3):491–538.

Oflazer, Kemal, Çetinoğlu, Özlem és Say, Bilge. Integrating Morphology with Multi-word Expression Processing in Turkish. In: Tanaka, Takaaki, Villavicencio, Aline, Bond, Francis és Korhonen, Anna szerk. *Second ACL Workshop on Multiword Expressions: Integrating Processing*, Barcelona, Spain. Association for Computational Linguistics, July, 2004, 64–71.

Pearce, Darren. Using conceptual similarity for collocation extraction. In: *Proceedings of the 4th UK Special Interest Group for Computational Linguistics*, 2001.

Pearce, Darren. A comparative evaluation of collocation extraction techniques. In: *Poceedings of the Third International Conference on Language Resources and Evaluation*, Las Palmas, Spain. 2002.

Pulman, Stephen G. The recognition and interpretation of idioms. In: Cacciari, Cristina és Tabossi, Patrizia szerk. *Idioms: Processing, Strucutre and Interpretation*, 11. fejezet. Lawrence Erlbaum Associates, Hillsdale, NJ, 1993.

Sag, Ivan, Baldwin, Timothy, Bond, Francis, Copestake, Ann és Flickinger, Dan. Multiword Expressions: A Pain in the Neck for NLP. In: *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics (CICLING 2002)*, Mexico City, Mexico. 2002, 1–15.

Schone, Patrick és Jurafsky, Daniel. Is Knowledge-Free Induction of Multiword Unit Dictionary Headwords A Solved Problem? In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Pittsburgh, PA. 2001.

Sciullo, Anna-Maria Di és Williams, Edwin. *On the Definition of Word.* MIT Press, Cambridge, MA, 1987.

Shimohata, Sayori, Sugio, Toshiyuki és Nagata, Junji. Retrieving Collocations by Co-occurrences and Word Order Constraints. In: *Proceedings of ACL-EACL 97*, 1997, 476–481.

Villavicencio, Aline, Copestake, Ann, Waldron, Benjamin és Lambeau, Fabre. Lexical Encoding of MWEs. In: Tanaka, Takaaki, Villavicencio, Aline, Bond, Francis és Korhonen, Anna szerk. *Second ACL Workshop on Multiword Expressions: Integrating Processing*, Barcelona, Spain. Association for Computational Linguistics, July, 2004, 80–87.

Váradi, Tamás. The Hungarian National Corpus. In: *Proceedings of the Second International Conference on Language Resources and Evaluation*, Las Palmas. 2002, 385–389.

Angol címek felismerése

Pohl Gábor[1], Ugray Gábor[2]

[1] Pázmány Péter Katolikus Egyetem, Információs Technológiai Kar
1083 Budapest, Práter utca 50/A
pohl@itk.ppke.hu

[2] MorphoLogic Kft.
1126 Budapest, Orbánhegyi út 5.
ugray@morphologic.hu

Kivonat: Angol szövegek szintaktikai elemzése során nehézséget jelentenek a címekre jellemző sajátos nyelvtani szerkezetek: hiányos névelőhasználat, egyes segédigék elhagyása stb. A probléma a szintaktikai szabályok paraméterezésével küzdhető le anélkül, hogy elfogadhatatlan kompromisszumokat kelljen hozni akár a címek lefedettségét, akár a jólformált mondatok pontosságát illetően. A szöveget alkotó szegmensek besorolására aszerint, hogy címről van-e szó vagy sem, egy döntési fát tanítottunk be weboldalakról gyűjtött, kézzel osztályozott két korpusz segítségével. Az egyik korpuszon tanítva és a másikon tesztelve a módszerrel 95% körüli pontosságot sikerült elérni. Az eredmény megfelelőnek bizonyult ahhoz, hogy beépítsük a MorphoLogicnál fejlesztett angol-magyar gépi fordítórendszerbe. A cikkben ismertetjük a nyelvtani probléma természetét, a tanításhoz használt korpuszokat, a relevánsnak bizonyult tulajdonságokat és a tanítási módszert, valamint annak értékelését.

1 Angol címek felismerése és elemzése

1.1 Szintaxis

Az angol címek szintaxisa jelentős mértékben különbözik a közönséges angol mondatokétól. Tekintsük az alábbi két, címben illetve folyó szövegben található előfordulást:

1. Driver killed in accident

2. A driver has been killed in an accident.

Jól látható, hogy a címben használatos formából hiányoznak a határozatlan névelők, a *has* segédige ragozott alakja, a passzív szerkezethez tartozó *be*, valamint a mondat végén nincs írásjel. A *driver* mint determinálatlan, egyetlen megszámlálható főnévből álló főnévi csoport a szabályok megengedőbbé tételével még lefedhető lenne, ez az irány azonban önmagában véve végzetes túlgenerálást eredményez a bonyolult, ámde jólformált mondatok esetében. A sikeres elemzéshez tehát meg kell különböztetni a címeket a folyó szövegtől.

1.2 Paraméterezhető nyelvtan

A MetaMorpho rendszer [2][4] nyelvtani elemzőjének [3] fejlesztésekor a problémára új, a fordítóprogramokban ismereteink szerint eddig nem alkalmazott megoldást dolgoztunk ki. Az ötlet azon a felismerésen alapul, hogy ha egy mondatról még a fordítás előtt el lehet dönteni, hogy cím-e, akkor utána a nyelvtan paraméterezhető ennek megfelelően.

A nyelvtan paraméterezése a gyakorlatban annyit jelent, hogy az elemzendő szövegről a nyelvtanon kívül kiértékelt információk feltételként befolyásolják egyes szabályok alkalmazását. Amennyiben például a felismerő a bemenetként kapott szövegről megállapította, hogy az cím, a fent leírt determinálatlan NP-t létrehozó szabály működése engedélyezett, ellenkező esetben nem.

1.2 *Dokumentumformátumok*

Modern dokumentumformátumok esetében azt várhatnánk, hogy a címeket valamilyen speciális módon megjelölik a szerzők. A gyakorlatban azonban még ha létezik is ilyen formátumfüggő jelölési lehetőség, annak használata nem konzisztens. A különböző dokumentumformátumokban fellelhető eltérő jelölések felismerése gyakorlatilag megvalósíthatatlan feladat a dokumentumtípusok sokfélesége és zártsága miatt, ráadásul elterjedt dokumentumtípusokban (pl. PDF, PostScript) nincsenek is ilyenek. Ezért amellett döntöttünk, hogy a formázatlan szöveg (plain text) alapján próbáljuk egy osztályozó segítségével megkülönböztetni a címeket és a nem címeket.

2 Korpuszalapú címosztályozás

2.1 Döntési fa

Szabályalapú osztályozó létrehozásához nem állt rendelkezésünkre elég ismeret ahhoz, hogy a címként fordítandó szövegrészeket a nem címként fordítandóktól meg tudjuk különböztetni, ezért egy gépi tanuló algoritmus korpusz alapú tanítása mellett döntöttünk. A lehetséges gépi tanuló módszerek közötti választáskor fontos szempont volt, hogy az offline tanított osztályozó működése átlátható, illetve a későbbiekben könnyen – és lehetőleg szabályalapú változatban – beépíthető legyen a fordítórendszerbe, azaz ne feketedobozként működjön. A döntési fák ismert rossz tulajdonságaik (pl. instabilitás [1]) mellett is a legjobb választásnak tűntek, mivel egyszerre képesek nominális és numerikus tulajdonságok (feature-ök) kezelésére, valamint az eredményül kapott döntési fa ember által könnyen olvasható, procedurális programozási nyelveken könnyen implementálható. Egy lecsupaszított (pruned) döntési fa alkalmazása mellett szólt, hogy nem láthattuk előre, a szöveg szegmenseiből esetlegesen kinyerhető tulajdonságértékek közül melyek fogják majd befolyásolni döntésünket, azaz a releváns tulajdonságok meghatározását is a döntési fa tanító algoritmusra bíztuk.

A kísérlethez a WEKA[3] gépi tanulórendszer [6] *J48* döntési fa osztályozóját választottuk, amely Quinlan C4.5 döntési fa tanuló algoritmusán [5] alapul.

2.2 A címek megkülönböztető tulajdonságai

A döntési fa tanításához meg kellett határoznunk, hogy a tanítóhalmazba az egyes szövegegységek milyen tulajdonságait vegyük fel, illetve hogy mit válasszunk szövegegységnek. Az utóbbi döntés esetében a mondat és a bekezdés közül a bekezdést választottunk vizsgálatunk alapjául, mivel megfigyeléseink azt mutatták, hogy a címek mindig külön bekezdésbe kerülnek, azaz nincsenek, vagy nagyon ritkák a cím és nem cím szövegrészeket vegyesen tartalmazó bekezdések. A bekezdésszint választása azzal az előnnyel is együtt járt, hogy nem kellett számolnunk a címek (azaz rendhagyó tulajdonságokat mutató mondatok) esetében ismeretlen pontossággal működő mondatszegmentáló modul hibáival. Vizsgálatunk alapjául így a bekezdések következő tulajdonságait választottuk:

- szavak száma;
- záró írásjel (illetve ennek hiánya);
- névelők aránya;
- *be* és *have* különböző alakjainak aránya;
- nagybetűvel kezdődő szavak aránya.

Első kísérleteinknél a fentieken kívül a mondatok gépi mondatszegmentáló segítségével meghatározott számát is rögzítettük.

2.3 Tanító és kiértékelő minták előállítása

Mivel a MetaMorpho fordítórendszer fő célkitűzései közé tartozik weboldalak angolról magyarra fordítása, egy weblapokból készített kisméretű korpuszt hoztunk létre a kísérleteinkhez. Két hírportálról (cnn.com és newyorker.com) gyűjtöttünk össze oldalakat, melyekből kinyertük a bekezdésként értelmezhető szegmenseket és minden bekezdés esetében kézzel meghatároztuk, hogy cím-e. A CNN korpusz 776, a NEWYORKER korpusz 1739 bekezdést tartalmazott. A korpusz bekezdéseihez az előző pontban részletezett tulajdonságokat meghatározva készítettünk a WEKA rendszerben alkalmazható ARFF formátumú mintahalmazt. Mindkét mintahalmaz több címet tartalmazott, mint nem címet. Ez az ilyen jellegű weboldalak tulajdonsága, hiszen minden oldalon több másikra hívják fel címek segítségével a figyelmet.

2.4 Tanítás és kiértékelés

Az egyes adathalmazokkal külön-külön illetve az adathalmazokat kombinálva is betanítottunk döntési fákat. Először a CNN korpuszon tanítottuk és teszteltük keresztkiér-

[3] Waikato Environment for Knowledge Analysis

tékeléssel (tenfold cross-validation) a döntési fát. Az eredmény meglepően jó volt, a CNN korpusz bekezdéseinek 96,7%-át helyesen osztályozta a keresztkiértékeléssel tanított döntési fa. A fát megvizsgálva meglepve tapasztaltuk, hogy csupán egyetlen, a bekezdések szószámban mért hosszára vonatkozó szabállyal 95%-os pontosság volt elérhető. A NEWYORKER korpuszon tesztelve a fát (most nem keresztkiértékeléssel tanítva) 90,6% volt a jó döntések aránya, csak egy szabályt alkalmazva viszont 91%-volt, ami az mutatja, hogy a döntési fa a CNN korpusz tulajdonságait túlzottan pontosan megtanulta, a NEWYORKER korpuszon így a kevésbé speciális egyetlen szabály jobbnak bizonyult.

A NEWYORKER korpusz már egy kicsit nehezebben klasszifikálhatónak tűnt. Keresztkiértékeléssel 93,7%-os osztályozási pontosságot ért el a fa, egyetlen, a bekezdés hosszára vonatkozó szabályt alkalmazva pedig 92,3%-ot. A CNN korpuszon tesztelve a fát 96,1% illetve 93% (csak egy szabályt alkalmazva) volt a helyesen osztályozott bekezdések aránya, ami igen jónak tekinthető.

A két korpusz unióján tanítva a fát az eredmény túl bonyolultnak tűnt. A paramétereket változtatva a fát sikerült redukálni, ugyanakkor azt tapasztaltuk, hogy a NEWYORKER korpuszon tanított jóval kisebb méretű fa is azonosan jó, 94,8%-os pontosságot ért el. A fa egyszerűsége miatt a túltanulás veszélye is jóval kisebb volt ebben az esetben, így a kísérlet végén ezt, az 1. ábrán látható fát találtuk legjobbnak.

Fontos megjegyezni, hogy minden döntési fánál a bekezdések szavakban mért hossza volt a legfontosabb döntési tényező; a többi tényező meglehetősen inkonzisztens módon változott a különböző fákat tekintve. Ez utóbb betudható a tanítóhalmazok különbözőségének, a döntési fa tanulás instabilitásának, illetve annak, hogy a tényezők súlya meglehetősen kicsi volt. A NEWYORKER korpuszon tanított fa esetében szerencsés, hogy a többi döntési tényezőhöz képest nehezebben meghatározhatók (a *be* és *have* különböző alakjainak aránya, a névelők, illetve a nagybetűs szavak aránya) estek ki a fa visszavágása (pruning) során.

```
wordCount <= 8: 1 (1133.0/70.0)
wordCount > 8
|   wordCount <= 18
|   |   endingMark = NONE
|   |   |   wordCount <= 11: 1 (30.0/3.0)
|   |   |   wordCount > 11: 0 (15.0)
|   |   endingMark = quest_m: 0 (19.0/4.0)
|   |   endingMark = excl_m: 0 (0.0)
|   |   endingMark = punct
|   |   |   wordCount <= 16: 0 (111.0/18.0)
|   |   |   wordCount > 16
|   |   |   |   wordCount <= 17: 0 (4.0/1.0)
|   |   |   |   wordCount > 17: 1 (3.0)
|   |   endingMark = colon: 0 (1.0)
|   wordCount > 18: 0 (423.0/4.0)
```

1. ábra: a NEWYORKER korpuszon tanított döntési fa

2.4.1 Következtetések

A kísérlet végén levonhattuk a következtetést, hogy az angol címek általában rövid bekezdések, rövidebbek 9 szónál, a nem cím bekezdések pedig hosszabbak. Utólag nem is tűnik különösnek ez az állítás, az viszont igen, hogy csupán ezzel az egy szabállyal 90% fölötti osztályozási pontosságot lehet elérni. Köszönhető ez annak, hogy ritkák az egyetlen rövid mondatot tartalmazó nem cím bekezdések, talán csak dialógusokban szerepelnek ilyenek, ezek elemzésére pedig – a címekhez hasonlóan hiányos szerkezetük miatt – lehet, hogy amúgy is szerencsésebb a címnyelvtant választani. A kísérlet eredménye arra is rámutat, hogy szerencsés választás volt a döntést bekezdésszinten meghozni.

2.5 A döntési fa alapú osztályozó modul alkalmazása

A döntési fa alapú címosztályozó C++ implementációját beépítettük a MorphoLogic MetaMorpho fordítórendszerébe, amelyhez a cím, illetve közönséges mondatok elemzésére és fordítására alkalmas alternatív nyelvtani szabályrendszereket Ugray Gábor és Merényi Csaba (MorphoLogic) dolgozták ki.

Referenciák

1. Li, Ruey-Hsia; Belford, Geneva G.: Instability of decision tree classification algorithms. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, 2002.
2. Prószéky Gábor; Tihanyi László: MetaMorpho: A Pattern-based Machine Translation Project. In: Proceedings of the 24th 'Translating and the Computer' Conference. London, United Kingdom, 19–24 (2002)
3. Prószéky Gábor; Tihanyi László; Ugray Gábor: Moose: A Robust High-Performance Parser and Generator. EAMT Workshop, Malta, 2004
4. Tihanyi László: A MetaMorpho projekt története. Magyar Számítógépes Nyelvészeti Konferencia 2003, Szeged.
5. Quinlan, J. Ross: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Francisco, 1993.
6. WEKA (Waikato Environment for Knowledge Analysis)
   Web page: http://www.cs.waikato.ac.nz/~ml/weka/index.html)
   Book: Witten, Ian H.; Frank, Eibe: Data Mining: Practical machine learning tools with Java implementations. Morgan Kaufmann, San Francisco, 2000.

V. Morfológia, szótár

Nyílt forráskódú morfológiai elemző

Németh László*, Halácsy Péter* Kornai András**, Trón Viktor***

Kivonat A cikk a Szószablya projekt keretében kifejlesztett nyelvfüggetlen morfológiai elemző keretrendszert mutatja be. A MorphBase rendszer a Myspell helyesírás-ellenőrző programkönyvtár továbbfejlesztése. Új tulajdonságai, a többszörös affixumleválasztás, a kimeneti információ, az alternatív tövek és elemzések lehetősége, és az összetettszó-kezelés, alkalmassá teszik bonyolult agglutináló nyelvek helyesírás-ellenőrzésére, tövezésére, és morfológiai elemzésére. Az algoritmusok futásidőben feldolgozott nyelvspecifikus célra optimalizált erőforrásokat használnak (tő- és affixumtár), amelyek karbantartását és generálását nagyban megkönnyíti az erre a célra kifejlesztett HunLex előfeldolgozó.

1. Bevezetés

A 2003-ban indult Szószablya projekt [4,6] egyik célja egy olyan nyílt nyelvtechnológiai eszközkészlet elkészítése volt, amely bővíthető, más rendszerekbe könnyen integrálható és szabadon felhasználható. A projekt műszakilag olyan elemzőalgoritmusok kidolgozását tűzte ki célul, amelyek támogatni tudják tetszőlegesen komplex agglutináló alaktannal rendelkező nyelvek elemzését. A rendszer teljes architektúráját az 1. ábra szemlélteti. A HunTools eszközrendszer három futtatható komponense a Hunspell helyesírás-ellenőrző, a Hunstem tövező és a Hunmorph morfológiai elemző. Ezek mindegyike és az ezek alatt futó MorphBase alapkönyvtár és így az egész HunTools eszközkészlet természetesen nyelvfüggetlen, így a nevükben szereplő *Hun-* előtag a fejlesztés eredetére, nem pedig a magyar nyelvű erőforrásokra utal. A keretrendszer alapját képező elemzőalgoritmusok az Ispell helyesírás-ellenőrző család által használt módszer továbbfejlesztett változatai. A rendszer implementációja az Ispell technológián alapuló helyesírás-ellenőrző függvénykönyvtár, a Myspell kódjának továbbfejlesztésével történt, és több szószintű elemzőalgoritmust tartalmaz: helyesírás-ellenőrzés, tövezés és morfológiai elemzés automatikus vagy interaktív hibajavítással. A számos új képességgel felruházott alapkönyvtár az általánosabb MorphBase nevet kapta.

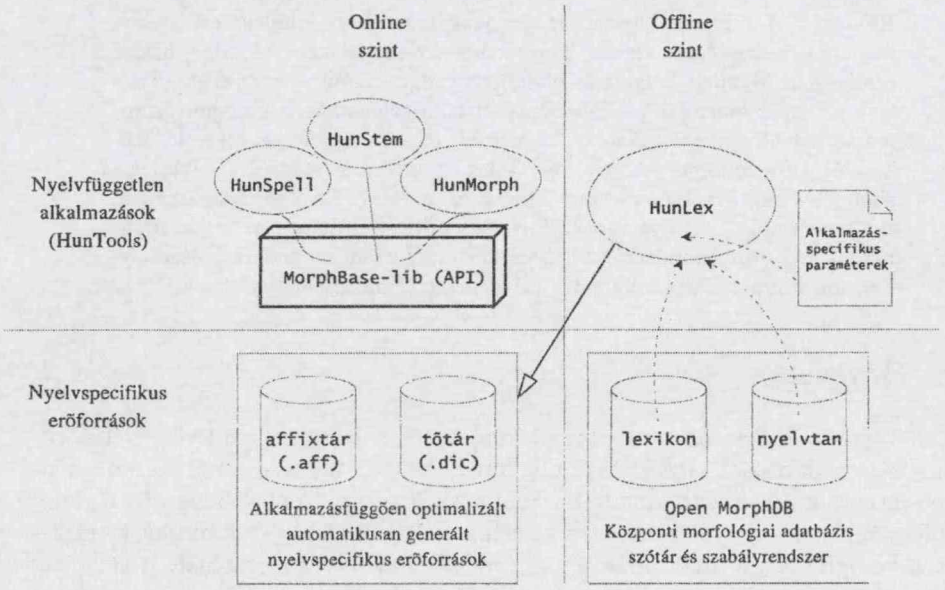
Az algoritmusok függvénykönyvtár formájában is elérhetők, amely a szoftvermodulokkal együtt a GNU LGPL licenc alatt szabadon felhasználhatóak.

* Budapesti Műszaki Egyetem Média Oktató és Kutató Központ, {nemeth,halacsy}@mokk.bme.hu

** MetaCarta Inc., andras@kornai.com

*** International Graduate College, Saarland University and University of Edinburgh, v.tron@ed.ac.uk

A Szószablya projekt kiemelt célja egy magyar nyelvű nyílt morfológiai elemző kifejlesztése volt. Az ehhez szükséges nyelvi erőforrások – magyar morfológiai szótár és szabályrendszer – előállítását és továbbfejlesztését nagyban képes segíteni a HunLex előfeldolgozó komponens [9]. Ennek a munkának az eredménye egy folyamatosan bővülő morfológiai adatbázis, amelyre épülő elemző rendkívüli lefedettséget és kielégítő pontosságú elemzéseket ad.



1. ábra. A Szószablya szóelemzési technológia felépítése

A nyelvtechnológia fejlődését mindig is komolyan befolyásolták a helyesírás-ellenőrzési alkalmazások. Ez a magyarban is így volt: a hétköznapi felhasználók számára fontos Helyes-e? jóval megelőzte a csak a szakértőknek érdekes HelyesLem és Humor alkalmazásokat [8]. A rendszer belső logikáját tekintve azonban azt látjuk, hogy ha megbízhatóbbá akarjuk tenni az elfogadás vagy elutasítás „egyszerű" dichotómiáján alapuló döntéseket, az alaktani tudás egyre alaposabb rendszerbe építésére van szükség. Jól mutatja ezt a Spell, Ispell, Myspell programcsalád fejlődése is.

A cikk további részében ismertetjük az Ispell technológia történeti hátterét és működésének alapjait (§2). Ezt követően tárgyaljuk a MorphBase algoritmusainak újításait (§3), valamint az egyes elemzési algoritmusok különbségeit (§4).

2. Affixumleválasztásos helyesírás-ellenőrzés

Első közelítésben a helyesírás-ellenőrzés leegyszerűsíthető egy gyakori és helyes szóalakokat tartalmazó halmazban való keresésre [3]. Az első helyesírás-

ellenőrzésre használható, Les Earnest által a hatvanas években fejlesztett SPELL program szótára is csupán a leggyakoribb tízezer angol szóalakot tartalmazta[2].

Nyilvánvaló, hogy kizárólag a lista bővítésével teljes lefedettséget nem lehet elérni, de a szótáron kívüli (out-of-vocabulary, OOV) elemek forrása inkább a produktív alaktanban keresendő: a jól formált összetett szavak és toldalékolt alakok egy morfológiailag komplex nyelv esetében végtelen halmazt alkothatnak.

Earnest tanítványa, Gorin 1971-ben nemcsak a szótárat bővítette tovább, hanem a hatékonyabb Spell helyesírás-ellenőrző programmal bevezette a – még heurisztikus, azaz nyelvtanilag motiválatlan – szuffixumleválasztást.

Az általunk is használt, a tövek és a hozzájuk kapcsolható affixumcsoportok elkülönült tárolása miatt az erőforrások méretének jelentős csökkenését eredményező *affixumleválasztást* (affix stripping) mint egyszerű elemzési módszert[7] egészítette ki Ackerman a tövekhez csatolható affixumokat kódoló ún. *affixumkapcsolókkal* (affix flags). A jelentősen átírt program 1978-ban Ispell (ITS version of Spell) néven látott napvilágot, amelynek változatai ma is széles körben használatosak, és amely a MorphBase fejlesztés kiindulópontjának tekinthető. Az Ispell szellemű elemzési módszer élesen szétválasztja a nyelvfüggetlen elemző algoritmust a nyelvspecifikus erőforrásoktól, amelyek az adott nyelv töveit és toldalékolási szabályait szöveges állományok formájában adják meg.

A tőtár egykarakteres affixumkapcsolókat tartalmaz a tőtől perjellel elválasztva, amelyek az affixum-szabályok alkalmazhatóságát vezérlő privatív lexikai jegyeknek is felfoghatók:

```
kár/A
jár/AB
```

Az affixumállomány pedig definiálja a kapcsolókhoz tartozó affixumokat:

```
SUFFIX A om
SUFFIX B tam
```

A fenti definíciók a két szótári szón kívül a *károm*, *járom* és a *jártam* alakokat is generálják.

A Geoff Kuening által 1998-ban bevezetett *affixumtömörítésnek* (affix compression) köszönhetően egy affixumkapcsoló nem csak egy affixumot, hanem affixumok tetszőleges halmazát jelöli, így az affixumkapcsolók immár részparadigmákat engedélyező jegyként interpretálhatók. Tételezzük fel, hogy affixumdefiníciónk a következő:

```
SUFFIX A ba
SUFFIX A ban
SUFFIX A ból
```

Ekkor, a szótárban szereplő *vár/A* definíció a *várba*, *várban*, *várból* alakot egyaránt engedélyezi. Ez az affixumtáblán alapuló megoldás nagyszámú toldalékolási szabály redundanciamentes kódolását teszi lehetővé, miáltal az Ispell technológia hatékonyan alkalmazható az angolnál lényegesen komplexebb morfológiájú agglutináló nyelvekre is.

Az affixum-szabályok alkalmazhatóságát további feltételekkel korlátozhatjuk: az affixumállományban minden szabályhoz megadható még egy illesztési feltétel, továbbá egy (a tő széléről) levágandó karaktersorozat. Az alábbi példában a harmadik mező az affixum illesztése előtt a tő végéről levágandó karaktersorozatot tartalmazza (vagy nullát).[1] A negyedik mező a tő végén alkalmazott illeszkedési feltételt tartalmazza :

```
SUFFIX B 0 val    [óúv]
SUFFIX B 0 zal    [^s]z
SUFFIX B z szal   sz
```

Az első szabály szerint az *ó*, *ú* és *v* karakterre végződő tövek (amelyeknek van B kapcsolója) megkaphatják a *val* szuffixumot. A második engedélyezi, hogy azok a *z* karakterre végződő tövek, amelyek utolsó előtti betűje nem *s*, *zal* szuffixumot kaphatnak. A harmadik szabály szerint pedig az *sz*-re végződő szavak *szal* toldalékot kapnak, de a toldalék illesztése előtt levágásra kerül a tő végi *z* karakter. A levágásokkal az egyszerűsítő írásmódot használó összetételek (*sz+sz* → *ssz*), illetve a „hasonulások" (*lát* → *lássa*) alternatív tövek felvétele nélkül valósíthatók meg.

3. A MorphBase fejlesztés összetevői

A MorphBase fejlesztése egy olyan nyílt helyesírás-ellenőrző fejlesztéséből indult ki, amely képes volt a magyar nyelv komplex agglutináló morfológiáját, valamint a magyar helyesírás bonyolult rendszerét kezelni [5]. A MorphBase kódjának alapja a Myspell könyvtár, amely az Ispell technológia C++ nyelvű implementációja.[2] A Myspell előnyei, (i) az affixumleválasztást felgyorsító indexelés [1], (ii) a nyelvspecifikus erőforrások futásidőben történő beolvasása, (iii) a szálbiztosság, és nem utolsósorban, (iv) a teljesen szabad licenc, mind hozzájárultak ahhoz, hogy a fejlesztés alapjául válasszuk. Ezek a tulajdonságok a teljes HunTools programcsomagot is jellemzik, ezzel lehetővé téve a kód hatékony integrálását és újrafelhasználását.[3]

A MorphBase fejlesztés közül a legfontosabbak az (i) homonim tövek kezelése, (ii) a többszörös affixumleválasztás bevezetése, valamint (iii) a parametrizálható összetettszó-kezelés. Ezeket az újításokat az alábbiakban részletesen tárgyaljuk.

3.1. Homonimák

A *homonimák* kezelése a Myspellben nincs megoldva: egy karaktersorozathoz (tőhöz) pontosan egy affixumkapcsoló-halmaz rendelhető. Ez problémát jelent

[1] Prefixum esetén a tő elejére, szuffixumnál a végére vonatkozik a levágás.

[2] A Myspell könyvtár az OpenOffice.org nyílt forráskódú irodai programcsomaghoz készült és a korábbi zárt kódú ellenőrző-modult váltotta ki.

[3] A HunTools programcsomag C++ nyelven íródott, és a nyílt forrású fejlesztésekben használt standard segédeszközök (Make, jelenleg Automake, Autoconf) biztosítják a program platformfüggetlenségét.

a pontos helyesírás-ellenőrzés számára, hiszen így a kategoriálisan többértelmű tövekhez több homonim tő (lemma) jegyeinek unióját kellett rendelni. Ezzel azonban az adott tőformához ellentmondó affixumok is illeszkedhetnek egyazon szóalakban. Például a helytelen **elöle* alak is elfogadásra kerül az *öl* ige okán engedélyezett igekötő és az *öl* főnév okán engedélyezett névszói toldalék egyidejű megjelenése miatt. A MorphBase esetében megvan a lehetőség a homonimák elkülönítésére, így az igei prefixumok és a névszói szuffixumok egyidejű megjelenését pusztán a szótári besorolások le tudják tiltani. A tőszótár tehát támogatja a homonimák megadását:

```
öl/P  # ige
öl/S  # főnév
```

Így az affixumállományban szereplő igei és főnévi szabályok, pl.:

```
PREFIX P el
SUFFIX S e
```

egyazon tőre nem alkalmazódnak.

3.2. Rekurzív affixumleválasztás

Az Ispell az elemzés során csak egy prefixum- és egy szuffixumszabály alkalmazását engedi meg. Egyetlen szuffixum leválasztása szűk lehetőséget kínál az olyan agglutináló nyelvek kezelésére mint a magyar, hiszen az összes számításba jövő toldalékmorf-kombinációt mind egy-egy affixumszabálynak kell megfeleltetni. A teljes magyar inflexiós rendszer húszezer körüli kombinációt jelent, a produktív derivációs toldalékolás miatt viszont egy főnévi lemmának akár $10^3 - 10^6$ alakja is lehet, az igekötőket is figyelembe véve pedig újabb két nagyságrenddel nagyobb számot kapunk.Az affixumkombinációk tárolása tehát komplex morfológia esetén praktikus problémaként merül fel. Ennek a megoldására a MorphBase elemző algoritmusa többszörös affixumszabály-alkalmazást is megenged. Ez annyit jelent, hogy elemzéskor az affixumleválasztással feltételezett hipotetikus tövekről újabb affixumok választhatók le. A többszörös affixumleválasztás jelenlegi megvalósítása ugyan nem teljes rekurziót, csupán kétszeres szuffixumleválasztást takar, gyakorlatilag viszont már ezzel is négyzetesen csökkenthető a szuffixumok száma, így az erőforrások mérete, vagyis végső soron a program memóriaigénye. A magyar nyelvi erőforrás affixumállományában így sikerült a produktív képzőket korlátozások nélkül leírni, miközben a képzett alakok elhagyásával a tőszótár mérete is jelentősen csökkent.

Ennek a bővítésnek köszönhető az is, hogy a prefixumok már nem csak a tövekhez, hanem az affixumokhoz is köthetők. Mivel az alkalmazásuk feltételeként megkövetelhetnek egy affixumot, valójában circumfixumok implementálását is lehetővé teszik. Például a magyarban circumfixumként kezelendő a melléknévi felsőfokot kifejező *leg–bb* toldalékegyüttes: A *leg-* prefixum csak akkor kapcsolódhat a tőhöz, ha a *-bb* (megfelelő alakja) is kapcsolódik (**legpiros*, *legpirosabb*). Az

ilyen circumfixumok kezelése eddig csak a szavak szótárban történő felsorolásával volt lehetséges.

A rekurzív affixumleválasztás implementálásával természetesen a nyelvspecifikus erőforrások formátumát is ki kellett bővíteni.[4] Az affixumdefinícióban két új mező jelenik meg. A kimeneti morfológiai információ (az elemző számára, 6. mező), illetve a „folytatási információ" (7. mező), amely az affixált alakra alkalmazható további affixumszabályok kapcsolóit tartalmazza.

```
PREFIX P   0 leg    .
SUFFIX R   0 ak     . [PLUR][NOM]
SUFFIX Q   0 bb     . [SUP]          PRN
SUFFIX Q   0 bb     . [COMP]         R
```

A példában a harmadik szabályban a P kapcsolók jelenléte engedélyezi a *leg-* prefixumot. Mivel a *leg-* prefixumra csak a *-bb* toldalékaffixumot tartalmazó minta hivatkozik, az első sor szabálya csak a harmadik sor alkalmazása „után" adhat helyes alakot és így az olyan alakok mint **legpiros* ki vannak zárva. A helyesírás-ellenőrzéshez már ennyi is elég volna, hiszen a *-bb* szuffixum viszont állhat a *leg-* nélkül (*pirosabb*), csakhogy a morfológiai elemzés számára ez nem állja meg a helyét, hiszen a *pirosabb* szóalak középfokú melléknévként elemzendő. A *-bb* toldalékmorf valójában homonim, és csak a felsőfok jelentésben „folytatható" a *leg-* prefixummal, sőt felsőfok jelentésben kötelező a *leg-* prefixum (e kölcsönös függés miatt nevezzük circumfixumnak). Azt, hogy egy szóalak csak tovább toldalékolva jelenhet meg, szintén egy kapcsoló adja meg (a harmadik sorban lévő N, ti. „nem tő"). Utóbbival tetszőleges olyan kötött tövek is megadhatók a szótárban, amelyek a továbbtoldalékolás szempontjából hasznosak, de nem jelenhetnek meg szabadon (pl. *lov*, *bokr*). A kötött tövek kezelését lehetővé tevő speciális kapcsoló szintén a fejlesztés során bevezetett újítás.

3.3. Összetett szavak kezelése

A számos további bővítés és új programparaméter közül az összetett szavak kezelését érdemes kiemelni. Az összetettszó-kezelés főbb tulajdonságai:

- megadható, hogy mely szavak szerepelhetnek szóösszetételben, akár csak az összetett szó első, vagy utolsó tagjaként, (a Hunspell esetében a legtöbb köznév ilyen, kivéve például a hónap- és napneveket),
- megadható a 6–3-as szabály (*kerékpárjavítással*, de *kerékpár-javítási*),
- megadható, hogy mely affixumok megléte esetén szerepelhetnek, illetve nem szerepelhetnek szóösszetételekben a képzővel ellátott szavak (*mérőléc*, de **mérendőléc*).

Tapasztalataink szerint a nyelvfüggetlen összetettszó-kezelés nem, vagy csak igen körülményesen valósítható meg: az összetett szavak helyesírása igen nehezen parametrizálható a nyelvi erőforráson keresztül. Ezért a forráskódban elkülönítve

[4] Ez a formátum kompatibilis a régi Myspell erőforrásokkal is.

jelenik meg egy az összetett szavak felismerését segítő osztály, lehetővé téve a modulnak a különböző nyelvekhez illetve célokhoz adaptált változatainak elkészítését.

4. Egy módszer, egy adatbázis, számos algoritmus

A helyesírás-ellenőrzéssel szemben a morfológiai elemzéshez vagy a tövezéshez nem elegendő a szóalakok elfogadásáról dönteni, hanem a releváns morfoszintaktikai kategóriák jelenlétét felismerve a bemeneti szóhoz kimenetként annotációt, tőindexet vagy lemmát kell rendelni. Emiatt az elfogadáshoz szükséges elemzéssel párhuzamosan a MorphBase elemző- és tövezőalgoritmusa a megfelelő kimenetkezelővel is kiegészül. A morfológiai elemzésnél minden szabályalkalmazás során az affixumhoz rendelt annotáció (kategóriacímke) regisztrálásra kerül, így sikeres elemzés esetén az algoritmus képes visszaadni a szó morfológiai elemzését.

A Myspell algoritmusa, akár a MorphBase helyesírás-ellenőrző algoritmusa az első elfogadott elemzés után nem keres továbbiakat, hiszen az elfogadáshoz ez nem szükséges. A tövezéshez és a morfológiai elemezéshez azonban a többértelműségek kezelése (nem a feloldásuk) alapvető követelmény, legyen az morf homonima (pl. *ár*, *fürdik*) vagy strukturális elemzési többértelműség (*értek*), így ezek az algoritmusok az összes alternatív elemzést meg tudják keresni. A teljes elemzés bevezetése egy helyen lehet korlátozva: a szavak összetett szóként való elemzését eszerint csak abban az esetben adja vissza a program, ha nem akad egyszerűbb elemzés. Például a *halász* elemzésénél nem kapjuk meg a szabályok szerint helyes *hal+ász* felbontást, mivel a szónak van más, nem összetett szavas elemzése is. Ez a módszer jól definiált nyelvfüggetlen szűrést eredményez, és tapasztalatunk szerint jelentősen csökkenti a felesleges többértelműségeket. Az olyan alakok, amelyeknek a homonim egyszerű elemzése mellett szükséges az összetett szóként való elemzése is (pl. *karóra*), a szótárba kerülnek, és így nem esnek áldozatul a korlátozásnak.

Az elemzés az affixumlevágásos módszerrel történik mindhárom szóelemző feladat esetében. A javaslattevés és hibajavítás képességétől elvonatkoztatva tehát az algoritmusok csak az alábbi két dimenzió mentén parametrizálhatók:

1. **elemzés teljessége**:
   (a) első elfogadott elemzésig (helyesírás-ellenőrzés, tövezés gyorsított indexeléshez)
   (b) korlátos többszörös elemzés
   (c) teljes elemzés (egyszerűbb elemzés esetén is adhat összetett szavas elemzést)
2. **kimenetfeldolgozás**:
   (a) nincs (helyesírás-ellenőrzés)
   (b) csak a tőtárban (tövezés)
   (c) mind a tövekhez, mind az affixum-szabály alkalmazáshoz (morfológiai elemzés)

A feladatokhoz szükséges minden egyéb konfiguráció a nyelvi erőforrások szintjén történik. Bár ez a feladat egyáltalán nem triviális, a HunLex előfeldolgozó segítségével egy közös adatbázisból automatikusan állíthatók elő a valósidejű alkalmazások számára különféleképpen optimalizált erőforrások.

5. Összefoglalás

Cikkünk általános tanulsága – ahogy ezt a MorphBase (a Hunspell, Hunstem, és Hunmorph programok alapja) is bizonyítja –, hogy a három legfontosabb szószintű elemzési feladat, a helyesírás-ellenőrzés, a tövezés, és a morfológiai elemzés egységes módszertannal kezelhető. Ez programozástechnikailag nem evidens, hiszen ugyanahhoz a bemenethez a helyesírás-ellenőrzés bináris (elfogad, elutasít) döntést, a tövező a tövet, a morfológiai elemzés pedig egy összetett, részint paradigmatikus (inflexió), részint szintagmatikus (deriváció) adat-struktúrát rendel. Az, hogy ezt a három problémát mégis egységes keretben érdemes kezelni, a tudományos közfelfogással is ütközik némileg, hiszen a helyesírás-ellenőrzést könnyű (lényegében listázással megoldható), míg a morfológiai elemzést nehéz, bonyolult algoritmusokat és nyelvészeti szakértelmet igénylő problémának szokás tekinteni.

A HunTools nyílt forráskódú programkönyvtárunk egyedülálló lehetőséget teremt magyar nyelvű szövegek elemzésére. A HunTools rendszer a GNU LGPL licenc alapján szabadon felhasználható, módosítható és nagyobb ipari rendszerekbe integrálható, akár önállóan futtatható szoftvermoduljait, akár a MorphBase függvénykönyvtár algoritmusait használva. A kompatibilis erőforrásformátumnak köszönhetően több, mint 40 nyelvhez használható helyesírás-ellenőrzőként, valamint szótövezőként (a morfológiai kimeneti annotáció hiányában morfológiai elemzőként nyilvánvalóan nem). Eszközünk alkalmazására különösen az Ispell technológia által mostohábban kezelt agglutináló nyelvek körében számítunk. Felhasználása mind kutatói, mind ipari körökben elkezdődött: a visszajelzések igazolják és egyben tovább növelik nyílt forrású fejlesztési modellünk sikerét.

Köszönetnyilvánítás

A Szószablya projekt az Informatikai és Hírközlési Minisztérium ITEM pályázatán nyert támogatással vált lehetővé. A program erőforrásait jelentős részben a MATÁV Rt. és az Axelero Internet biztosította. Fejlesztéseinkhez a Szószablya fejlesztőkön kívül a Magyar Ispell és a Szószablya levelezőlisták olvasóinak észrevételei is hozzájárultak. Segítségüket mindannyiuknak köszönjük.

Hivatkozások

1. B Dömölki. Algorithms for the recognition of properties of sequences of symbols. *USSR Computational & Mathematical Physics*, 5(1):101–130, 1967. Pergamon Press, Oxford.

2. Les Earnest. Machine recognition of cursive writing. *Information Processing*, 1963. Proc. IFIP Congress 1962, Munich.
3. Péter Halácsy, András Kornai, László Németh, András Rung, István Szakadát, and Viktor Trón. Szógyakoriság és helyesírás-ellenőrzés [word frequency and spell-checker accuracy]. In *Proceedings of the 1st Hungarian Computational Linguistics Conference*, pages 211–217. Szegedi Tudományegyetem, 2003.
4. Péter Halácsy, András Kornai, László Németh, András Rung, István Szakadát, and Viktor Trón. Creating open language resources for Hungarian. In *Proceedings of Language Resources and Evaluation Conference (LREC04)*. European Language Resources Association, 2004.
5. Németh László. Magyar Ispell – válasz a Helyes-e?-re. In *IV. GNU/Linux szakmai konferencia*, pages 99–107. Linux-felhasználók Magyarországi Egyesülete, 2002.
6. Németh László. A Szószablya fejlesztés. In *V. GNU/Linux szakmai konferencia*. Linux-felhasználók Magyarországi Egyesülete, 2003.
7. James Lyle Peterson. *Computer programs for spelling correction: an experiment in program design*, volume 96 of *Lecture Notes in Computer Science*. Springer, 1980.
8. Gábor Prószéky and László Tihanyi. Humor – a morphological system for corpus analysis. In *Proceedings of the first TELRI seminar in Tihany*, pages 149–158, Budapest, 1996.
9. Trón Viktor. Hunlex - morfológiai szótárkezelő rendszer. In *II Magyar Számítógépes Nyelvészeti Konferencia*, 2004.

Általános célú morfológiai elemző kimeneti formalizmusa

Kornai András*, Rebrus Péter**, Vajda Péter* Halácsy Péter***, Rung András**, Trón Viktor†

Kivonat Az alábbi írásban egy a szóalakok morfoszintaktikai ábrázolására használható formalizmust mutatunk be. A formalizmus alapvető adatstruktúrája egy speciális fagráf, amely mind inflexiós mind derivációs információ megragadására alkalmas. A fagráfból rekonstruálható egy inflexiós információt tartalmazó teljes bináris jegy–érték-struktúra, ugyanakkor redundanciamentes és egyszerű linearizálhatósága folytán jól alkalmazható egy általános célú morfológiai elemző kimeneti kódrendszereként. Nyelvészeti megalapozottsága alkalmassá teszi arra is, hogy a szótári tételeket morfoszintaktikai viselkedését megjelenítse egy morfológiai adatbázisban.

1. Bevezetés

Egy morfológiai elemző kimeneti formalizmusának három, egymásnak gyakran ellentmondó feltételt kell kielégítenie. Ezek az *informativitás:* a lehető legpontosabban és legteljesebben tükrözze a szóalakokból megállapítható morfológiai információkat; *adekvátság:* nyelvészetileg megalapozott kategóriákat használjon; *egyszerűség:* kézi, illetve automatikus feldolgozásra egyaránt könnyen használható legyen. Mindhárom feltétel nagy mértékben függ a felhasználó céljaitól: minél pontosabban határozzák meg az elemzés célját (pl. helyesírás-ellenőrzés, tövezés, szintaktikai elemzés, korpuszalapú statisztikai vizsgálatok), annál könnyebben lehet megfelelő egyensúlyt találni köztük. Egy előre nem specifikált célú morfológiai elemző esetében azonban nincs mód arra, hogy a kimeneti annotáció tökéletes legyen. Cikkünk azt kívánja bemutatni, hogy a Szószablya projekt([2]) keretében elkészült HunTools szóelemző eszköztár ([4]) kimeneti kódrendszerének megalkotásakor a fenti feltételeket hogyan próbáltuk meg összehangolni.

Az alábbiakban először az inflexiós toldalékolás ábrázolására használt hierarchikus struktúra alapelveiről szólunk. Bemutatjuk, hogy ez a gazdag struktúra hogyan linearizálható és egyszerűsíthető a morfológiai jelöltség felhasználásával (§2). Ezek után ismertetjük a magyar morfológiai elemző kimenetében használt

* MetaCarta Inc., e-mail: andras@kornai.com
** MTA Nyelvtudományi Intézet, e-mail: {rebrus,vajda}@nytud.hu
*** BME Média Oktató és Kutató Központ {hp,runga}@mokk.bme.hu
† IGK, Saarland University, University of Edinburgh v.tron@ed.ac.uk

főnévi és igei inflexiós kategóriarendszert, röviden kitérve a derivációkezelésére is (§3). Végül megemlítjük, hogy a formalizmus alkalmas különböző kódolást alkalmazó morfológiai elemzők összehasonlítására is (§4).

2. Az inflexió fagráfos ábrázolása

Egy szóalak morfológiai ábrázolása úgy teljes, ha az összes inflexiós tulajdonsága specifikálva van. Az inflexiós tulajdonságok túlnyomó része a mondattani elemzésben játszik fontos szerepet, szintaktikai szabályok hivatkoznak rájuk. Az ilyen morfoszintaktikai tulajdonságok szokásosan ún. jegy–érték-struktúrákkal (attribute-value structure, AVS) [5] adhatók meg. Ez a struktúra a szóalak morfoszintaktikai vetületét hivatott ábrázolni és mint ilyen független az azt kódoló formai jegyektől és a szóalak felszíni ábrázolásától. Meggyőződésünk, hogy az a morfoszintaktikai annotáció, amely elvonatkoztat a morfszegmentálástól (amely valójában a morfológia item-and-arrangement felfogásához kötődik), elméletsemlegessége és modularitása miatt szélesebb körű alkalmazást tesz lehetővé.

A szóalak morfológiai jegyei nem homogének; két jellemzőjüket kell kiemelnünk: (i) hierarchikusság: bizonyos jegyek specifikálása más jegyek jelenlétét, specifikálását feltételezi, és (ii) aszimmetria: adott jegy lehetséges értékei közül bizonyosak jelöletlennek, mások jelöltnek tekinthetők.

Ezt a két jellemzőt jól megragadja egy címkézett fagráf. A fagráf csomópontjai az inflexiós jegyek (címkéi a jegyek nevei), a fa gyökere pedig a szótári tételek inflexió szempontjából vett ekvivalenciaosztályai (az inflektálható kategóriák). A fában jelenlévő csomópontok egy a gyökér-csomópont által meghatározott jegyösvény pozitív értékét jelentik. Ez egyben azt is jelenti, hogy a gráf kizárólag olyan bináris jegy–érték-struktúrát képes kódolni, amelyben csak a pozitív értékű csomópontoknak lehetnek a fában folytatásai [3]. Ez a megszorítás a jelöltség fogalmának egy értelmezése. A fagráfos ábrázolás a fenti követelményeket mind kielégíti: (i) informatív, hiszen jegy–érték szerkezetek formájában képes ábrázolni a szóalakok releváns morfoszintaktikai tulajdonságait; (ii) adekvát, hiszen megragadja az inflexiós információ hierarchikus aspektusát és a morfológiai jelöltség fogalmait; valamint (iii) egyszerű, hiszen belőle egy teljes bináris jegy–érték-struktúra automatikusan rekonstruálható, ugyanakkor redundanciamentessége miatt tömören linearizálható.

3. A főnévi és igei alakok inflexiós kódrendszerei

A magyar nyelv morfológiai elemzéséhez a fenti formalizmusban egy konkrét jegyrendszert dolgoztunk ki. A főnevekhez az alábbi hierarchikus struktúrát rendelhetjük, ami egyben a `NOUN` gyökércsomópontból kiinduló gráfok szignatúrájának felel meg.

Szám:	egyes	<-PLUR>
	többes	
	„egyszerű" (pl. *sógorok*)	<+PLUR<-FAM>>
	familiáris birtokos (pl. *sógorék*)	<+PLUR<+FAM>>
Birtokos:	nincs megjelölt birtokos	<-POSS>
	van, ekkor a birtokos	
	Személye	
	1. (pl. *sógorom*)	<+POSS<+1><-2>>
	2.	<+POSS<-1><+2>>
	3.	<+POSS<-1><-2>>
	Száma	
	egyes (pl. *sógorai*)	<+POSS<-PLUR>>
	többes	<+POSS<+PLUR>>
Birtok:	nincs birtok	<-ANP>
	van; a birtok száma:	
	egyes (pl. *sógoré*)	<+ANP<-PLUR>>
	többes (pl. *sógoréi*)	<+ANP<+PLUR>>
eset:	„nincs" (= nominativus)	<-CAS>
	van, 16 különböző eset lehet (pl. *sógort*)	<+CAS<+ACC>>

Egy inflektált alak morfoszintaktikai annotációja tehát egy fagráffal adható meg. A fagráfban a NOUN-ból kiinduló rész-ösvények a jegy-mátrix pozitív értékeit kódolják. A szignatúra által adott többi releváns jegy értéke mind negatív. A gráf tehát ekvivalens egy a szóalak inflexiós tulajdonságait leíró teljes bináris jegy–érték-struktúrával. A fagráf ábrázolás azonban redundanciamentes és a csomópontcímkék (attribútumnevek) zárójelezésével karakterlánckénti egyszerűen linearizálható, ennélfogva szöveges formában is egyszerűen és tömören kezelhető. Az alábbi példák a szóalak teljes inflexiós specifikációját mutatják jegy–érték-struktúrával, valamint a linearizált fagráfos kódolással.

kutya

```
<+NOUN<-PLUR><-POSS><-ANP><-CAS>>
<NOUN>
```

kutyáink

```
<+NOUN<+PLUR<-FAM>><+POSS<+1><-2><+PLUR>><-ANP><-CAS>>
<NOUN<PLUR><POSS<1><PLUR>>>
```

kutyáéi

```
<+NOUN<-PLUR><-POSS><+ANP<+PLUR>><-CAS>>
<NOUN<ANP<PLUR>>>
```

kutyáikéit

```
<+NOUN<+PLUR<-FAM>><+POSS<-1><-2><+PLUR>><+ANP<+PLUR>><+CAS<+ACC>>>
<NOUN<PLUR><POSS<PLUR>><ANP<PLUR>><CAS<ACC>>>
```

Amint látható, a hierarchikus szerkezet lehetővé teszi, hogy egyazon primitív jegy (csomópontcímke) különböző dependensek konceptuálisan azonos morfo-

szintaktikai jegyeinek kódolására is legyen használható, így például a főnév, a birtokos, illetve a birtok többes száma egyaránt a PLUR jegy használható.

A jelöletlen morfoszintaktikai jegyértékeket a legtöbb esetben zérusmorfémák fejezik ki. Ez azzal az előnnyel jár, hogy a kódolás nagyjából tükrözi a szóalakokban található testes morfok számát (vagy az alkalmazott toldalékolási operációk számát), anélkül hogy állást foglalna a morfológiai szegmentálás (a felszíni szóalak toldalékallomorfokra bontásának) kényes kérdésében, tehát informatív, de ez nem megy az adekvátság rovására.

A hierarchikus szerkezet linearizálásában használt zárójelezési konvenció segítségével élesen elhatárolható az alulspecifikáció a teljes inflexiós specifikációtól. A <NOUN> teljesen specifikált (egyes számú, nem birtokos, nominatívuszi) alakot jelöl, míg a NOUN lexémák egy osztályát, ami mint felszíni alak minden inflexiós jegyre alulspecifikált. Formálisan a zárójelezett forma egy modellt a zárójel nélküli pedig egy modellhalmazt ír le. Ezzel az ábrázolás alkalmas a szótári tételek morfoszintaktikailag releváns paradigmatikus információjának ábrázolására. (pl. *kutya* NOUN). Ez a szófaji információ közvetlenül kompatibilis a szótári tétel inflektált alakjának az elemző által kiadott kódjával, ugyanakkor képes kifejezni a lehetséges inflektált alakokra vonatkozó megszorításokat a szótárban. Azokat a szótári tételeket, melyek paradigmája (morfológiai okokból) hiányos, az inflexiós fagráf részleges specifikációjával adhatjuk meg a szótárban. Például:

- plurale tantum (*üzelmek, üzelmei, üzelmeim*, stb., vs **üzelem*, **üzelme* stb.)
  üzelmek NOUN<PLUR>
- possessivum tantum (*eleje, elejem, elejei*, stb. vs. **ele/elő*, **elék/elők* stb.)
  eleje NOUN<POSS>
- possessivum és plurale tantum (*elei, eleim* vs. **el/ele/elő*, **elek/elők*, **ele/eleje/elője*, stb.)
  elei NOUN<PLUR><POSS>

Képzés. A fentebb bemutatott fagráfok közvetlenül nem alkalmasak a szóképzés ábrázolására. Ugyanakkor a képzőket tekinthetjük szótári tételek közötti relációkként. Így a derivációs viszonyok az inflexiós ekvivalenciaosztályok (a fagráf lehetséges gyökercímkéi, pl. NOUN, VERB, ADV) közötti címkézett irányított élekkel ábrázolhatók, ami valójában az inflexiós gráfstruktúra kiterjesztése. Kimeneti kódolásukra a következő néhány példa adható:

faxol <[fax]NOUN[1]VERB>
faxolgat <[fax]NOUN[1]VERB[gAt]VERB>
faxolgatás <[fax]NOUN[1]VERB[gAt]VERB[Ás]NOUN>

Az eddigi megfontolásokból automatikusan adódik, hogy sem az inflektált alakok, sem a részben specifikált (azaz nem gyökér) „tantumok" nem vethetők alá képzésnek.

4. A morfológiai elemzők kimeneti kódjainak megfeleltetése

Mivel ez az ábrázolás független a morfológiai elemzés technológiai megvalósításától, alkalmas arra, hogy több különböző morfológiai elemző kimeneti formalizmusának közös nevezője lehessen. A morfológiai elemző kimeneti kódrendszerének tervezésekor megvizsgáltuk többek között az MSD-kódrendszert [1]. A magyar nyelvre fent ismertetett kódot úgy alakítottuk ki, hogy az legalább annyi morfoszintaktikai információt tartalmazzon, mint az MSD-kódok, így az utóbbiak átalakítása egyértelmű legyen. Ehhez elkészítettünk egy transzformációs táblázatot, amely a kialakítandó kódokra való leképezést adja meg. Ennek segítségével a Szószablya projekt keretében elkészített magyar morfológiai elemzőt össze lehet vetni más kódolást használó rendszerekkel, s ezzel a kiinduló morfológiai adatbázisunk hibái és hiányai egyszerűbben javíthatók.

Hivatkozások

1. T. Erjavec and M. Monachini. Specifications and notation for lexicon encoding. Technical report, Copernicus Project 106 MULTEXT-East, December 1997.
2. Péter Halácsy, András Kornai, László Németh, András Rung, István Szakadát, and Viktor Trón. A szószablya projekt. In *Proceedings of the 1st Hungarian Computational Linguistics Conference*. Szegedi Tudományegyetem, 2003.
3. András Kornai. A főnévi csoport egyeztetése. In Telegdi and Kiefer, editors, *Általános Nyelvészeti Tanulmányok*, XVII. Akadémiai Kiadó, Budapest, 1989.
4. László Németh, Viktor Trón, Péter Halácsy, András Kornai, András Rung, and István Szakadát. Leveraging the open-source ispell codebase for minority language analysis. In *Proceedings of SALTMIL 2004*. European Language Resources Association, 2004.
5. Viktor Trón. Attribútum-érték struktúrák. In László Kálmán, Viktor Trón, and Károly Varasdi, editors, *Lexikalista elméletek a nyelvészetben*. Tinta Könyvkiadó, Budapest, 2002.

Hunlex - morfológiai szótárkezelő rendszer

Trón Viktor*

Kivonat Cikkünkben[1] a HunLex szótárkezelő és morfológiai erőforrásgeneráló keretrenszert mutatjuk be. A HunLex lehetővé teszi, hogy egy könnyen fenntartható, átlátható de gazdagon specifikálható központi nyelvi adatbázisból kiindulva szószintű elemzőalkalmazások erőforrásait állítsuk elő. A HunLex prototípusa a Szószablya fejlesztés keretében megvalósított HunTools szóelemző eszköztár moduljai számára készít optimalizált nyelvspecifikus erőforrásokat, de elméletileg kész más rendszereket is kiszolgálni. A kimeneti erőforrások számos paraméter mentén igény szerint konfigurálhatók.

1. Bevezetés

A Szószablya projekt [4] legközvetlenebb célja egy nyílt magyar nyelvű morfológiai elemző kifejlesztése volt. Az ehhez szükséges nyelvi erőforrások – magyar morfológiai szótár és szabályrendszer – előállítását és továbbfejlesztését nagyban képes segíteni a HunLex előfeldolgozó komponens. A Szószablya szóelemző technológia [9,8] felépítését a 1. ábra szemlélteti.

A HunLex bemenete egy szakértői munkával létrehozott és fenntartott központi nyelvi adatbázis, kimenete pedig a valósidejű alkalmazások által közvetlenül értelmezhető erőforrás. Látható, hogy akárcsak a MorphBase elemző függvénykonyvtár rutinjai, úgy a HunLex is nyelvfüggetlen rendszer, amely két nyelvspecifikus morfológiai adatbázis közötti konverziót hivatott elvégezni.

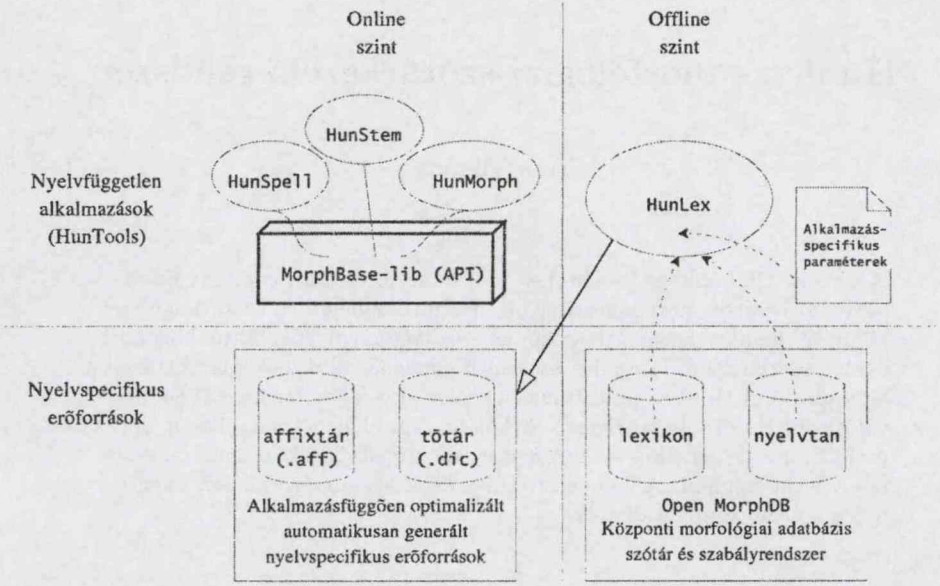
A cikk további részében ismertetjük a HunLex keretrendszert. Elsőként a HunLex elkészítésének motivációját tárgyaljuk (§2), majd röviden bemutatjuk a jelenlegi renszer fontosabb jellemzőit (§3). Végül a HunLex rendszer lehetséges további felhasználási lehetőségeit és a modul kiterjesztésére irányuló terveinket ismertetjük (§4).

2. Motiváció

Kényelmes bővíthetőség és fenntarthatóság. Alapvető elvárás, hogy egy valósidejű elemzőalkalmazás (helyesírásellenőrző, morfológiai elemző) nyelvfüggetlen legyen és az elemzéshez szükséges nyelvspecifikus tudást erőforrások formájában

* International Graduate College, Saarland University and University of Edinburgh, v.tron@ed.ac.uk

[1] Ezúton szeretnék köszönet mondani Halácsy Péternek, Konrai Andrásnak, Németh Lászlónak, Rung Andrásnak, Rebrus Péternek és Anne Benoitnak.



1. ábra. A Szószablya szóelemzési technológia felépítése

lehessen megadni. Az elemzés minőségét az erőforrásban megadott morfológiai és lexikai információ lefedettsége és pontosságát határozza meg. Emiatt nagyon fontos, hogy ezt az erőforrást könnyű legyen bővíteni és javítani. Az elemző hatékonyságának biztosítása érdekében azonban a nyelvi erőforrások formátuma gyakran nem alkalmas emberi feldolgozásra. Például a HunTools moduljainak erőforrásai bár szöveges állományok, igen redundánsak és nehezen áttekinthetőek, közvetlen szerkesztésük majdhogynem lehetetlen. Egy bonyolult morfológiájú, agglutinatív nyelv esetén az MorphBase affixumállománya számos toldalékmorf kombinációjából előálló affixumcsoportokat tartalmaz. Ha egy affixum viselkedését szeretnénk megváltoztatni, akkor az azt tartalmazó összes kombinációt figyelembe kell vennünk. Ez a feladat csak egy olyan keretrendszer segítségével végezhető szisztematikusan, amely lehetővé teszi a morfológiai szabályok és a lexikai elemek toldalákolási információinak rugalmas és következetes javítását.

Mindebből következik, hogy az elemzőalkalmazások erőforrásait érdemes offline automatikusan előállítani miközben az adatbázisok javítása és fenntartása átlátható központi formátum használatát igényli [1]. A kétféle szintű erőforrás között egy konfigurálható előfeldolgozó rendszer közvetít, egy ilyen keretrendszer mára a legtöbb elemzőtechnológiának része, így például a magyar nyelv elemzésére leginkább használt Humor rendszernek is [7].

Futásidejű elemzés hatékonysága. Mivel az elfogadás szempontjából nem fontos, hogy mit tekintünk tőnek illetve affixumnak, a helyesírás-ellenőrzőben az affixumok és tövek pontos meghatározása csak hatékonysági kérdésként merül fel. Egyes nyelvészetileg összetett alakok (általában a kivételes vagy improduktívan

toldalékolt alakok) felsorolással lehetnek kezelve, valamint a tő (itt: szótárban felsorolt sztring) fogalma nem azonos a lemma, vagy tőallomorf nyelvészetileg releváns fogalmával (például a *számat* „töve" *szám*, a *sarkam* „töve" pedig a *sarkak* a Magyar Ispell szótár eredeti állományaiban).

Hasonlóan, a kimeneti annotáció megvalósításához mind a tövek, mind az affixumcsoportok morfológiai annotációját meg kell adni. Gyakran előfordul, hogy a futásidejű elemzéskor használt tő–affixum felbontás nem feleltethető meg a kategóriák azonosítását szolgáló (és általában a morfológiai leírásként szolgáló) komponensekre-bontásnak.

Egyrészt számos imporduktív és kivételes alak a szótárban van felsorolva (pl. hatékonysági megfontolásokból), amelyeknak a morfológiai elemzését a lexikon kell, hogy kódolja. Másrészt egy affixumcsoport is potenciálisan tetszőleges számú morfológiailag releváns morf kombinációja lehet, ezért ezek „elemzését" is előre kódolnunk kell.

Az ilyen praktikus megfontolások azonban nem szabad, hogy befolyásolják a morfológiai elemzés kimenetét, vagyis az elemzés kimenete és az elemzés futásidejű implementációja ideális esetben függetlenítendő. Ugyanakkor a morfológiai adatbázis formátumát lehetetlen az egyes elemzési technológiák igényeihez optimálizni.

Algoritmusfüggő erőforrásoptimalizálás. Bár a helyesírás-ellenőrző tekinthető mint a morfológiai elemző egy leegyszerűsítése: ha a bemeneti szóalakhoz sikerül elemzést rendelni, akkor a szó helyes –, a kétféle elemzést hatékonyabb más módszerrel megoldani. Ugyanez igaz az információ-visszakereső (information retrieval) rendszerekben gyakran alkalmazott szótövező viszonyában, hiszen a tövek visszaadása során ugyan kezelni kell a tövek többértelműségét de például az egy kategorián belüli affixumtöbbértelműséget nem (irreleváns, hogy a *fürdik* alak 3SG-INDEF vagy 3PL-DEF). Egyértelmű tehát, hogy különböző elemzőrutinokhoz más és más erőforrás az optimális, előállításukat azonban érdemes egy központi adatbázisból automatikusan végezni.

Rugalmas alkalmazásfüggő erőforrásgenerálás. Az erőforrások alkalmazásfüggőségére további példa lehet, hogy egy morfológiai elemzőtől nagyobb rugalmasságot várunk el az akadémiai helyesírási szabályzat követésében, mint egy helyesírás-ellenőrzőtől (például hasznos, ha elemzi a gyakori **izület*, **lőjjünk*, vagy **adatbáziskezelő* szóalakokat is). Hasonlóan egy indexelésre használt szótövezőnél nem feltétlen hasznos, ha a szófaj-, illetve jelentős értelemváltozással járó képzések tövét adja vissza (például a *Sorstalanságról* töveként a *sors*-ot), ugyanakkor más feladatokhoz ez a tőinformáció hasznos lehet. Fontos szempont tehát, hogy egy központi adatbázisból szigorú, illetve engedékeny elemzők is előállíthatók legyenek, vagyis az erőforrásgenerálásnál lehetőséget kell adni az alul-, ill. túlgenerálásra.

3. Mit tud a hunlex?

Mindezen kívánalmak figyelembevételével terveztük meg a HunLex rendszert. A hunlex egy központi (gazdag információtartalmú) morfológiai adatbázisból dolgozik, de hogy pontosan milyen kimeneti erőforrást (a HunTools esetében ún. dic, illetve aff állományokat) kompilálunk, az számos szempont szerint változtatható.

Bemeneti források. A Hunlex konkrétan kétféle forrásból dolgozik: (i) a bázislexikon és nyelvtan a nyelv lexikonát és morfológiáját írja le; (ii) a többi állomány a kimeneti erőforrások kompilálását szabályozza.

A nyelv morfológiáját leíró hunlex lexikon és nyelvtan egyszerűen és átláthatóan specifikálható, így a folyamatos szótárbővítés és a morfológiai szabályok finomítása kényelmesen végezhető. A nyelvtanírást és a lexikon karbantartását segítik az egyszerűen definiálható makrók, amelyek reguláris kifejezésekhez is használhatók toldalékolási szabályok alkalmazási feltételeinek megadásához. Mivel lehetőség van a teljes nyelvtan és lexikon által generált nyelv előállítására, ezért a rendszerszerű tesztelés és a morfológiai leírás korábbi állapotaival való összevetés könnyen elvégezhető.[2]

Az erőforrásgenerálást vezérlő opciók beállításával a kimenet számos paraméter mentén konfigurálható.

- Állítható, hogy a kimenet helyesírás-ellenőrzés, tövezés, illetve morfológiai elemzés számára optimalizált dic illetve aff állományokat állítson elő.
- Kiválasztható, hogy mely toldalékolási szabályokat alkalmazza az elemző. Ezen belül megválasztható, hogy az elemző mely morfológiai szabályokat fogja alkalmazza futásidőben. Egyes morfológiai szabályok kompiláláskor alkalmazódnak a bázislexikon elemeire, így egyes morfológiailag komplex alakok is bekerülhetnek az elemző tőtárába. A hunstem tövező a tőtárból kikeresett tőinformációt adja vissza az elemzéskor, így ezzel az opcióval különböző mélységű tövezőket lehet kapni.
- Az futásidőben elemzendő toldalékmorfémák másik morfémákkal kombinálódhatnak és az eredményül kapott ún. affixumcsoportokat az elemző egy toldalékként (egy lépésben levágva) elemzi. Hogy mely toldalékok alkossanak csoportokat, azt a nyelvtantól függetlenül konfigurálható ún. szintek segítségével.
- Az egyes morfémaváltozatokat szabályozó morfofonológiai jegyek közül melyeket vegye figyelembe a rendszer. Bizonyos jegyek (részleges) kizárásával robusztus túlelemző nyelvtanok állíthatók elő.
- Korlátozható továbbá a rekurzív szabályalkalmazás mélysége.
- A morfológiai szabályok és a tövek különböző regiszter, ill. stílusjegyekkel lehetnek ellátva, amelyeket a kompilálás során figyelembe vesz a rendszer. Így például a helyesírásellenőrző számára szigorú normatív, egy robosztus elemző számára pedig hiperengedékeny forrás generálható.

[2] A hunlex morfológiai nyelvtant leíró formalizmusról és a specifikáció technikai részleteiről lásd a `http://www.szoszablya.hu` weboldalt.

- A kimeneti annotáció (tövező és morfológiai elemző számára) számos paraméter mentén konfigurálható. Többek között a hunlex képes beépített jegy-érték struktúrák kezelésére és unifikálására, ami igen rugalmassá képes tenni mind a kimeneti annotáció alakítását, mind a morfoszintaktikai kategóriák lexikai specifikációját [5].

A hunlex rendszer alkalmazása különösen hasznos olyan nyelvek leírására, amelyekhez szóelemző technológia nem áll rendelkezésre. Mivel a hunlex képes előállítani a megfelelő optimalizált erőforrásokat a nyílt licenszű HunTools csomag elemzőalgoritmusai számára, egyetlen egységes hunlex alapállomány segítségével akár ipari alkalmazásokba is beépíthető ellenőrző-, tövező- és morfológiai elemzőmodulok nyerhetők az adott nyelvre.

4. Lehetséges kiterjesztések

További erőforrás-formátumok. Bár a hunlex elsődlegesen a MorphBase szóelemző eszközkönytrár algoritmusainak kiszolgálására készült, egy intelligens szótárkezelőtől elvárható hogy további futásidejű elemzőprogramok bemeneti erőforrásait is képes legyen előállítani. Ilyen például a véges állapotú technológiát használó SFST, illetve XSFT. Jelenleg is folyik annak a vizsgálata, hogy a hunlex formalizmusban leírt morfológiai nyelvtanok hogyan kompilálhatók a fenti programok által használt erőforrások formátumára. Amennyiben a formalizmusok ereje kompatibilisnek bizonyul, várható, hogy a jövőben a hunlex ezeket az nyelvi erőforrásokat is képes lesz előállítani, illetve a különböző nyelvtanformalizmusok közötti konverziót elvégezni. Ezzel egyrészt biztosítható, hogy a hunlex nyelvtanokkal leírt nyelvek más elemzőkkel is használhatók legyenek. Másrészt, így a véges állapotú modellel leírt nyelvtanokat az affixumlevágással dolgozó MorphBase algoritmusai is megértik, és az adott nyelvekre rögtön helyesírásellenőrző- és tövező-alkalmazásokat is kapunk.

Szintén tervezés alatt van a hunlex lexikonoknak szabványos XML kódolásra való átalakítása. Ezzel a lexikai adatbázis portabilitása biztosítható, ami elősegíti a szótári információ szélesebb körben való használhatóságát. Erre felkészülve a hunlex alapszótárban már jelenleg is lehetséges az elemzőrutinok által nem használt információ felvétele tetszőleges attribútumok bevezetésével.

Nyílt magyar morfológiai adatbázis. A BME Médai Oktató és Kutató az MTA Nyelvtudományi Intézetének munkatársaival közösen egy nyílt magyar morfológiai szótári adatbázis fejlesztésén dolgozik. A leírás keretéül a hunlex szolgál. A hunlex lehetővé teszi, hogy az nagy lefedettségű és naprakész Magyar Ispell szótárat [6] összevessük az Akadémiai Nagyszótárral (pontosabban az Értelmező Kéziszótárban önálló címszóval szereplő szókinccsel, amely Papp Ferenc Debreceni Tezauruszán keresztül digitális formában szabadon elérhetővé vált [3]), valamint a Magyar Ragozási Szótárral [2]. Ezeknek az adatbázisoknak a kritikus összefésülésével az eddigi legteljesebb magyar morfológiai nyelvtan és szótári adatbázis készülhet el és válhat szabadon elérhetővé. A HunLex keretrenszer

biztosíték arra, hogy a szótári adatbázis nagy lefedettségét és pontosságát a HunTools programcsomag szóelemző moduljai kihasználhassák és így leíró célja mellett az adatbázis közvetlenül a magyar nyelvtechnológia hasznára lehessen.

Hivatkozások

1. I. Aldezabal, O. Ansa, B. Arrieta, X. Artola, A. Ezeiza, G. Hernández, and M. Lersundi. Edbl: a general lexical basis for the automatic processing of basque. In *IRCS Workshop on linguistic databases. Philadelphia*, pages 1–10, 2001.
2. László Elekfi. *Magyar ragozási szótár*. MTA Nyelvtudományi Intézet, Budapest, 1994.
3. Mihály Füredi, András Kornai, and Gábor Prószéky. A szótár adatbázis. Kézirat, 2004.
4. Péter Halácsy, András Kornai, László Németh, András Rung, István Szakadát, and Viktor Trón. A szószablya projekt. In *Proceedings of the 1st Hungarian Computational Linguistics Conference*. Szegedi Tudományegyetem, 2003.
5. András Kornai, Péter Rebrus, Péter Vajda, Péter Halácsy, András Rung, and Viktor Trón. Általános célú morfológiai elemző kimeneti formalizmusa. In *II. Magyar Számítógépes Nyelvészeti Konferencia*, 2004.
6. Németh László. Magyar Ispell – válasz a Helyes-e?-re. In *IV. GNU/Linux szakmai konferencia*, pages 99–107. Linux-felhasználók Magyarországi Egyesülete, 2002.
7. Attila Novák. Milyen a jó humor? In Zoltán Alexin and Dóra Csendes, editors, *Magyar Számítógépes Nyelvészeti Konferencia*, pages 138–144. Szegedi Tudományegyetem, Informatikai Tanszékcsoport, Szegedi Tudományegyetem, 2003.
8. László Németh, Péter Halácsy, András Kornai, and Viktor Trón. Nyílt forráskódú morfológiai elemző. In *II. Magyar Számítógépes Nyelvészeti Konferencia*, 2004.
9. László Németh, Viktor Trón, Péter Halácsy, András Kornai, András Rung, and István Szakadát. Leveraging the open-source ispell codebase for minority language analysis. In *Proceedings of SALTMIL 2004*. European Language Resources Association, 2004.

A Ragozási szótártól a NooJ morfológiai moduljáig

Vajda Peyter, Nagy Viktor, Dancsecs Erzsébet

MTA Nyelvtudományi Intézet, Budapest VI. Benczúr u. 33. Pf. 701/518 H-1399
{vajda,nagyv,mano}@nytud.hu

Kivonat A cikk a *NooJ* számítógépes nyelvészeti fejlesztőkörnyezet magyar morfológiai moduljának fejlesztését mutatja be. A morfológiai elemző alapja Elekfi László Magyar ragozási szótárának a Nyelvtudományi Intézetben megvalósított számítógépes változata. A morfológia leírásában a paradigmatikus ábrázolás helyett át kellett térnünk a jegy alapú osztályozásra. Az implementálásban a *NooJ* véges állapotú technikáját használtuk.

1. Bevezetés

A Nyelvtudományi Intézet Korpusznyelvészeti osztálya azt a célt tűzte ki, hogy megvalósítja az *Intex*, illetve utódja, a *NooJ* nyelvészeti fejlesztőkörnyezet (továbbiakban: *Intex/NooJ*) alá a magyar nyelvi modult, vagyis azokat az alapnyelvtanokat, amelyekre majd építhet a rendszer leendő felhasználói közössége. A dolgozat által bemutatott morfológiai elemző és lemmatizáló ennek a modulnak lesz a része.

2. Az Elekfi-rendszer

Az általunk kidolgozott morfológiai elemző alapját Elekfi László *Magyar ragozási szótára* ([2]), illetve az egyelőre csak kéziratban létező, a ragozási szótárnál jóval részletesebb *Szókincsünk nyelvtani alakrendszere* ([3]) című munkája adta. Az anyag gépre vitelével morfológiai adatbázis épült.A ragozási szótár címszóanyaga megegyezik a *Magyar Értelmező Kéziszótár* első kiadásának címszóanyagával. A morfológiai adatbázis tőtára a lexémák tőallomorfjait tartalmazza.

Az Elekfi-féle rendszer a lexémákat paradigmaosztályokba sorolja. Az osztályok kétdimenziós elrendezést mutatnak. Az egyik dimenziót az elölségi, illetve a kerekségi harmónia adja (betűjelek az osztálykódokban: A: „mély" (hátulképzett), B: „magas" (elölképzett), kerekítetlen, C: kerekített; igéknél kisbetűvel jelölve), a másik dimenziót a lexéma tövének egyéb komplex tulajdonságai adják. A második dimenzió az igéknél tizenkilenc, a névszóknál harminchat csoportot határoz meg. Ezek a csoportok finom paradigmatikus különbségek szerint további alcsoportokra oszlanak.

Ez a fajta paradigmaosztályozás több hátránnyal bír. A paradigmák közötti számos megegyezés és rendszerszerű különbség rejtve marad, hiszen a két dimenzió csak két tulajdonságot képes ábrázolni (bővebben lásd [5]). A paradigmák közötti finom különbségek pedig megnövelik az alcsoportok számát, hiszen minden egyedi ragozási sor egyedi osztályt kell, hogy alkosson. A teljes rendszer több mint 1700 paradigmaosztályt tartalmaz.

Ilyen sok osztály karbantartása és implementációja nehézkes. Egy megfelelő véges állapotú eszköz képes lehet optimális, redukált automata előállítására, azonban mind az *Intex*, mind a *NooJ* csak korlátozottan rendelkezik ilyesfajta képességekkel. A rendszer inflexiós modulja a redukálást generálással oldja meg: a szabályrendszer alapján előállítja az összes szótári szó lehetséges alakjait, melyeket automatába tömörít. Ez a megoldás megfelelő a szegényes inflexiót felmutató nyugat-európai nyelveknek, a magyarnak semmiképp. Még ha mellőztük is a rekurziót a magyar morfológiából, a néhány tízezres szótár több tízmillió lehetséges szóalakjának tömörítése erőforráskorlátokba ütközött.

Az Elekfi-rendszer további korlátja, hogy szűk a számba vett toldalékok köre. A paradigmák az inflexiós toldalékok mellett csupán néhány produktív képzőt tartalmaznak (igenévképzők, *-hAt*, *-Ás*, műveltető, *-(j)Ű*). További képzők hozzáadása a meglévővel ortogonális osztályozást igényel, hiszen gyakran olyan szemantikai tulajdonságoktól függ, hogy egy képző hozzájárulhat-e egy adott tőhöz, amelyekhez nem rendelhetők Elekfi-paradigmák.

3. A morfológiai leírás módszere

Mint az előbbiekben láttuk, a paradigmatáblák nehézkesen kezelhetők, nagyfokú redundanciát tartalmaznak. A kétdimenziós osztályozás miatt több paradigma tartalmazhat hasonlóan viselkedő elemeket, de az ilyen hasonlóságok csak akkor ragadhatók meg, ha ezen elemeket bármilyen szempont szerint be tudjuk sorolni egy-egy csoportba.

Ennek elérése érdekében jegyeket alakítottunk ki, melyek segítségével egymástól független szempontok szerint tudtuk a szavakat osztályozni. Majd felépítettünk egy, a fenti jegyeket használó véges állapotú transzducert. Ezzel a paradigmaosztályok fent említett hátrányait kiküszöböltük. Az alábbiakban részletesebben foglalkozunk az *Intex/NooJ* rendszer eszközkészletével (bővebben lásd [6]), valamint a transzducer felépítésével, példákon keresztül megmutatva a felmerült nehézségek megoldását.

3.1. Az Intex végesállapotú technológiája

Az *Intex* többek között szótárak, a derivációs és inflexiós morfológia, és a szintaxis leírásához, valamint korpuszok feldolgozásához nyújt eszközöket.

Az *Intex* a feldolgozás valamely pontján minden erőforrást (szótárak, nyelvtanok, stb.) végesállapotú transzducerrel (*Finite State Transducer, FST*) reprezentál. Egyszerűbb végesállapotú automaták leírására alkalmazhatunk reguláris kifejezéseket, bonyolultabb nyelvtanok beviteléhez pedig felhasználói felületként

egy gráf-építő modult használhatunk. Egy gráf egy FST-t reprezentál, amely esetünkben a (jólformált) bemeneti szöveghez morfológiai információt társít. Egy transzducer azonban nemcsak úgy fogható fel, hogy egy karaktersorozathoz egy másikat rendel, hanem úgy is, mint egy karaktersorozatot elfogadó véges állapotú automata (bővebben lásd [1]). Ez megfelel a morfológiai elemző feladatának, mely egyszerre elfogad és elemez egy szóalakot.

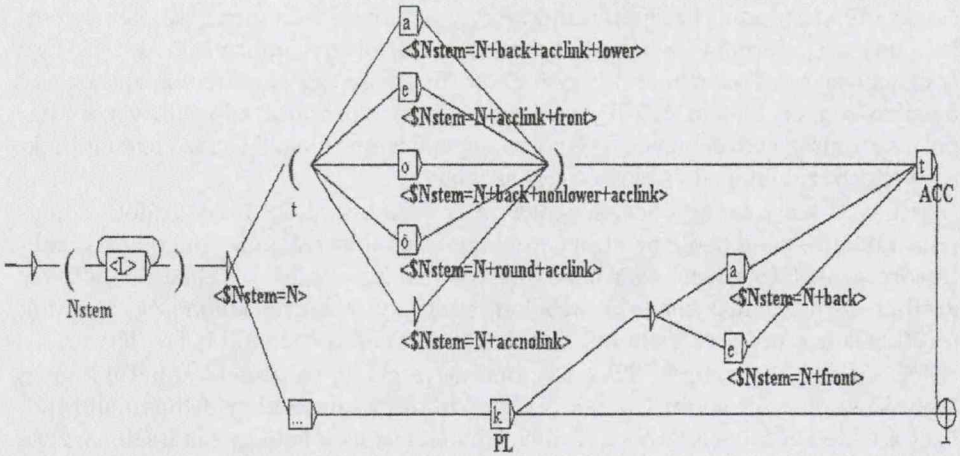
Ellentétben a transzducerek szokásos gráf alakjával, az Intex-gráfok csomópontjaikban – nem pedig az átmenetekben – tartalmazzák a felismerendő karaktersorozatokat (betűket) és a hozzájuk tartozó kimenetet, továbbá állapotok a gráfban nem jelennek meg. Ez azonban csak egy technikai különbség: a gráfok fordításakor a program minimalizált determinisztikus automatát hoz létre.

Az *Intex* kiterjesztett FST-ket is használ: ezekben változókat adhatunk meg, melyek az elemzés során kapnak értéket, majd a kimenetben felhasználhatjuk őket a bemenet módosításához. Ennek többek között a kétjegyű mássalhangzóra végződő szavaknál van szerepe, ahol egyes toldalékok esetében a tő és a toldalék konkatenációjával nem áll elő a helyes alak (pl.: *lány+nyá* ≠ *lánnyá*), ezért a bemenetből egy karaktert törölni kell.

Jegyalapú morfológia. A jegyalapú morfológiák sajátossága, hogy egy-egy (relatív) tő és (relatív) toldalék jegyeinek unifikációjával, a jegyek ellenőrzésével áll elő a toldalékolt alak. Az *Intex/NooJ* eszközeivel azonban csak a szótő és az első toldalék jegyeit lehet közvetlenül egyeztetni, az egymást követő toldalékokét már nem, mivel a toldalékok a szótárban nem szerepelhetnek, így ott jegyeket sem lehet hozzájuk rendelni. A toldalékokhoz rendelt megszorításokat pedig a szótárban kell ellenőrizni (lásd a 3.1. pontot). Ez az oka annak, hogy azokban az esetekben, ahol egy toldalékot relatív tőként egy másik toldalék követ, a gráfban új csomópontokat kell felvennünk a kapcsolódó toldalékhoz.

Ez a jelenség megfigyelhető például a többesszám és a tárgyeset kapcsolódásánál, amit az 1. ábra illusztrál. A tárgyeset viselkedésének leírását a gráfunkban két esetre bontottuk. Az első eset az, amelyben a tárgyeset közvetlenül az abszolút szótőhöz kapcsolódik. Ekkor a tárgyrag a *-(V)t* alakot veheti fel, ez a gráfban öt útvonalat jelent, amelyeket a megfelelő kötőhangokat kiválasztó jegyek engedélyeznek. A második esetben, nyitótőként viselkedő toldalékok után a tárgyeset alakja viszont már *-At*. Az *Intex/Nooj* eszközeivel nem tudunk olyan feltételt megfogalmazni, mely megadná, hogy a többesszám kötőhangjától függően miyen kötőhangot vár a tárgyeset.

Lexikai megszorítások. Az *Intex/NooJ* rendszerben lexikai megszorítások adhatók meg a szóalakok különböző részeire. A megszorítások lexikai jegyek formájában jelennek meg, és a szótár segítségével kerülnek ellenőrzésre. A lexikai megszorításokban hivatkozhatunk az elemzés során előzőleg definiált változók értékeire. Ennek segítségével oldottuk meg a szótő és a toldalékok elválasztását (lásd az 1. ábrát). A gráfban egy hurok inkrementálisan egy `Nstem` változóhoz rendeli a bemeneti sztring betűit, majd, hogy mi kerül végül szótőként ebbe a



1. ábra. A tárgyeset kezelése

változóba, azt a `<$Nstem=N>` megszorítás dönti el, amely a szótő meglétét ellenőrzi a szótárban (természetesen ezután az automatának még el kell jutnia a végállapotba).

Továbbá, ha a transzducer egy csomópontjában az `ö/<stem=N+round>` információ szerepel, akkor a gráfnak ezen az ágán csak azok a szavak tudnak továbblépni, amelyek a feldolgozás ezen pontján egy *ö* betűt tartalmaznak, és a szótárban a `+round` jeggyel szerepelnek. Az, hogy mind az *Intex*ben, mind a *NooJ*-ban csak egy jegy meglétét lehet ellenőrizni, annak hiányát nem, azt jelenti, hogy csak egyértékű jegyek adhatók meg. Ezért nem a (morfofonológiában) megszokott kétértékű jegyeket használtuk, hanem jegy-párokat (pl.: `+round`, `+unround`)

.

3.2. A jegyek kialakítása

Az általunk használt jegyek kialakításánál [5]-re és az Elekfi-rendszerre támaszkodtunk. Természetesen felhasználtuk a rendszer egyik dimenzióját adó, a magánhangzó-harmónia elemzéséhez szükséges jegyeket. Ezen túlmenően a kétdimenziós csoportok másik dimenzióját adó összetett tulajdonságokat (pl.: „zárt kötőhangzós főnevek puszta tárgyraggal"), valamint e csoportokon belül található rendszeres különbségtételeket (pl.: a *j* hang megjelenése a névszók harmadik személyű birtokos alakjánál) is feldolgoztuk. Ezek a tulajdonságok okozzák az Elekfi-rendszer redundanciáját és bonyolultságát, ezért ezekből egymástól független jegyeket alakítottunk ki, amelyek már alkalmasak egy többdimenziós osztályozás felállítására.

A tőallomorfia kezelése. A létrejött jegyek közül külön említést érdemelnek a tőallomorfiára vonatkozóak. Bár egy transzducer elvben képes szimbólumokat

törölni vagy beszúrni az elemzendő karaktersorozatba, ily módon kezelve a tőallomorfiát, ezt a kérdést az Elekfi-rendszerhez hasonlóan tőallomorfok használatával oldottuk meg. A morfológiai adatbázisban lévő töveket az ismert morfofonológiai osztályokba (pl.: hangkivető, rövidülő, bővebben lásd [4]) soroltuk, és a szótárban erre utaló jegyeket társítottunk hozzájuk. A szótárba a lexémák mellett a kötött tőallomorfok is bekerültek az erre utaló jegyekkel. Ezután megvizsgáltuk, hogy az allomorfok hogyan viszonyulnak az egyes toldalékokhoz. A különböző típusú tőváltozat-osztályokat tovább csoportosítottuk az alapján, hogy mely toldalékokhoz járulnak a szótári alakjukal (BASE), melyekhez pedig allomorfjukkal (OBL). A névszói tövek csoportosításának egy részlete látható az 1. táblázatban.

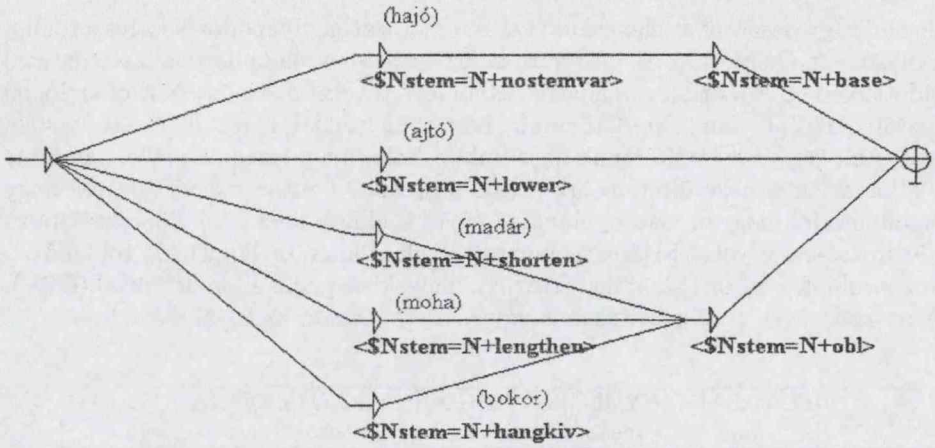
	EGYALAKÚ *hajó*	NYÚLÓ *moha*	RÖVIDÜLŐ *madár*	HANGKIVETŐ *bokor*	HANGVÁLTÓ *ajtó*
PS[et][12], ACC, PL.NOM, DIS/SOC,	BASE	OBL	OBL	OBL	BASE
PS[et]3	BASE	OBL	OBL	OBL	OBL/BASE
SUP	BASE	OBL	BASE	OBL	BASE
esetek, POS	BASE	OBL	BASE	BASE	BASE
NOM	BASE	BASE	BASE	BASE	BASE

1. táblázat. Tőosztályok

A táblázatból leolvasható, hogy ezen öt névszói tőosztály esetében öt csoportba kell beosztanunk a toldalékokat. A megvalósítás szintjén ez annyit jelent, hogy minden egyes csoporthoz egy-egy beágyazott gráfot készítettünk, mely az adott toldalék(ok)hoz kapcsolódik, és az összes tőtípus erre vonatkozó viselkedését írja le – azt, hogy a szótári alakkal, vagy pedig a kötött tővel jár-e együtt. Például a harmadik személyű birtokos alak algráfjában (lásd a 2. ábrán) a nem változó tőtípusok a szótári alakot (`+base`) kapják, a hangváltó tövek esetében mindkét allomorf lehetséges (*ajtója*, *ajtaja*, ill. *ajtói*, *ajtai*), ezért ide nem kerül megszorítás, a többi tőtípus pedig a nem szótári alakkal (`+obl`) jár együtt.

Természetesen további tőosztályok, rendhagyó paradigmák felvételével a toldalékok csoportosítása is megváltozik.

Rendhagyó ragozású tövek. Az Elekfi-rendszer 9-es csoportjai a rendhagyó szavakat tartalmazzák, amelyek az általános paradigmákba nem illeszthetőek be. Ezért ragozásuk leírása több nehézségbe is ütközik. Egyrészt a szótárban a szabályos paradigmákhoz kialakított jegyek alkalmazása nem elégséges, új jegyek felvétele viszont túlságosan bonyolulttá tenné a szótár és a főgráf szerkezetét. Megvizsgálva a csoportba tartozó főneveket, három lehetőség állt elő. Készíthetünk algráfokat, megpróbálhatjuk beilleszteni a töveket szabályos paradigmákba, vagy statisztikai okok miatt figyelmen kívül is hagyhatjuk őket.



2. ábra. A tőallomorfia kezelése

Általában véve ezeknek a töveknek a viselkedését célravezető algráfokban leírni, majd beilleszteni őket a főgráf szerkezetébe. Ebben az esetben csak azt a jegyet kell kiosztanunk, amely a rendhagyó paradigmát meghatározza, ugyanakkor alkalmazhatjuk a már meglévő jegyeket is pl. a hangrendre vonatkozóan. A tő viselkedése szempontjából például azonos, de hangrendileg különböző a 9A3 (*bátya*) és 9B3 (*néne*).

Helyenként alkalmazható eljárás a tövek már meglévő paradigmákba történő besorolása is. Például az egyetlen csoportba sorolt *fiú* főnév esetében szétválasztottuk belőle a *fi* és *fiú* töveket, amelyek már beilleszthetők egy-egy szabályos paradigmába. Vagy például a *h*-ra végződő tövek szintén beleillenek egy-egy szabályos paradigmába, viszont két esetragnál opcionálisan más paradigma toldalékait is megkaphatják (*cseh-vel*, de **cseh-hel*, ill. *méh-hel*, *méh-vel*), ezért ez a tőallomorfiánál látott módszerrel oldható meg.

A 9-es csoport nem minden alcsoportja került besorolásra hatékonysági szempontok miatt. Megvizsgáltuk, mely paradigmákhoz hány szó tartozik, illetve azt, hogy az alapszó milyen előfordulást mutat az MNSZ-ben. Azokat a paradigmákat hagytuk el, amelyek alá egyetlen szó tartozik, és a Magyar Nemzeti Szövegtárban nem fordul elő (ilyen például a *moholy* szó). Ezekre a perifériikusnak minősíthető tövekre nem készült külön algráf, illetve nem kerültek besorolásra. Hasonlóan jártunk el a régiesnek tekinthető szavakkal is (pl.: *tereh*).

Figyelembe vettük viszont azokat a paradigmákat, amelyekbe ugyan csak egy vagy két szó tartozik, viszont több előfordulást találtunk rá – ilyen pl. a *kehely*, mely csak alanyesetben 86-szor fordul elő az MNSZ-ben -, vagy intuitíve fontos szónak tartottuk ragozott alakjai miatt (mint pl. a *bátya* szó, mely ugyan csak 39-szer található meg az MNSZ-ben, viszont birtokos személyjeles alakjai jóval gyakoribbak, mert pl. a *bátyja* alak is ebből a paradigmából áll elő). Bár a

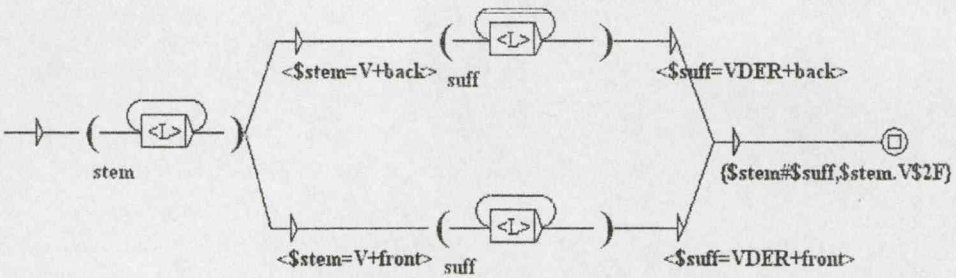
rendhagyó igék kimerítő vizsgálatára még nem került sor, valószínűsíthető, hogy a főneveknél alkalmazott eljárásokat itt is hasznosíthatjuk majd.

3.3. A szótár

A morfológiai elemző modul szótárát az Elekfi-rendszer szótárából alakítottuk ki. Az eredeti szótárban a számunkra lényeges információ a szótövek mellet a szófaj, az esetleges tőallomorfok és az illető szónak az Elekfi-rendszerbeli paradigmakódja volt. Az általunk használt szótárban a szótövek és a tőallomorfok külön bejegyzésként szerepelnek a szófaj és a szóra vonatkozó jegyek feltüntetésével. A paradigmakódok közvetlen jegyekre alakítása csak nehezen, kézzel lett volna elvégezhető, ezért más módszert választottunk. Mivel az Elekfi-adatbázisból a rendszer bármely szavának bármely ragozott alakja kinyerhető, ezért minden olyan alakot meg tudtunk vizsgálni, amely alapján eldönthető, hogy az egyes jegyek szempontjából, hogy viselkedik az adott szó. Például annak eldöntéséhez, hogy egy szó megkapja-e a `+acclink` jegyet (ami azt jelzi, hogy a tárgyesetben van-e kötőhangja), elegendő megvizsgálni a tárgyesetű alakját. Ily módon automatikusan az összes paradigmához hozzá tudtuk rendelni a viselkedését meghatározó jegyhalmazt.

4. Képzés

Fentebb láttuk, hogy a nemterminális toldalékoknak a tőlük jobbra álló morfémával szemben támasztott jegyigényeit nem tudjuk megfogalmazni, helyette új csomópontokat és utakat kell felvenni a gráfban. Emiatt a képzők felvételével sokszorosára növekedne az elemzőgráf bonyolultsága. Ehelyett azt tervezzük, hogy a képzőket fiktív tőként vesszük fel a szótárba. A 3. ábrán látható gráf felel a szótő és a képző összekapcsolásáért, ha a képzőnek csak egy magas és egy mély allomorfja van. A *suff* változóba kerülő részsztring már a képző továbbtoldalékolt alakja.



3. ábra. A képzés

5. Összefoglalás

A dolgozatban felvázoltunk egy fejlesztés alatt álló morfológiai elemzőt *Intex/NooJ* alá, amelyet terveink szerint hozzáférhetővé teszünk bárki számára nonprofit, kutatási célokra. A fejlesztőeszköz nem ideális a feladatra, hiszen elsősorban nyugat-európai nyelvekre készült. A magyar modul létrehozásának szándéka azonban ösztönzőleg hat az *Intex/NooJ* fejlesztőire, akik igyekeznek bővíteni a rendszer szolgáltatásait, hogy alkalmas legyen a magyar morfológia implementálására is.

A fejlesztés során szakítanunk kellett az Elekfi-rendszerrel, és jegyalapú osztályozást kellett alkalmaznunk. Ezáltal rugalmasabb, könnyebben karbantartható adatbázist kaptunk, amely alkalmas lehet más eszközökkel történő morfológiai elemző létrehozására is.

Hivatkozások

1. Dienes Péter: A fonológia modellezése végesállapotú eszközökkel. In: Lexikalista elméletek a nyelvészetben. Szerk.: Kálmán László, Trón Viktor, Varasdi Károly. Tinta Könyvkiadó, Budapest 2002. 17-51.
2. Elekfi László: Magyar ragozási szótár. MTA Nyelvtudományi Intézete, Budapest 1994.
3. Elekfi László: Szókincsünk nyelvtani alakrendszere. Kézirat. 1986.
4. Nádasdy Ádám és Siptár Péter: A magánhangzók. In: Strukturális magyar nyelvtan 1. Fonológia. Szerk.: Kiefer Ferenc. Akadémiai Kiadó, Budapest 1994. 42–182.
5. Prószéky Gábor: A magyar morfológia számítógépes kezelése. In: Strukturális magyar nyelvtan 3. Morfológia. Szerk.: Kiefer Ferenc. Akadémiai Kiadó, Budapest 2000. 1021–1063.
6. Silberztein, Max: The Intex Manual. `http://intex.univ-fcomte.fr/downloads/Manual.pdf`

Szófaji beosztás névszói csoportok elemzéséhez

Naszódi Mátyás: MorphoLogic, 1126 Budapest, Orbánhegyi út 5.
naszodim@morphologic.hu

Kivonat: A kiasszikus nyelvosztályozás – főnév, melléknév, számnév stb. – nem elégséges a gépi elemzéshez. A névszók pontosabb kategorizálása lehetőséget ad a többértelműségek számának csökkentésére, és pontosabb mondatelemzésre. Jelen cikk egy finomabb, de jelentéstant nem használó felosztást javasol. A felosztás szerepet játszik a szóalaktanban és a mondattanban egyaránt. Segítségével pontosítható a névszói szerkezetek határa és definiáltsága, mely hasznosnak bizonyult a korábbi projektekben, de használata várhatóan a fordítási projektben elengedhetetlenné válik.

1. Szófaji osztályozás

A szófaji osztályozás, mint kategorizálás két célt szolgál. Egyrészt a szótani szabályok megadásánál játszik szerepet, másrészt a mondattani szabályok ezekre, mint alapelemekre épülnek. A szófaji osztályok nyelvenként eltérnek. Ennek ellenére nyelvészeink, mint más népek nyelvészei is igyekeznek hasonló osztályokat, kategóriákat létrehozni, hagyomány szerint a latin grammatikát alapul véve. Ez hasznos, ha össze akarunk vetni különböző nyelveket, vagy nyelvtanulóknak ad jó támpontot. A gépi feldolgozás szempontjából is hasznos. Sajnálatosan vagy szerencsére a nyelvek különbözőek. Egyesek szerint angol nyelvben kisebb a különbség az ige és a névszó között. A szó aktuális szófaját nem a szó, mint inkább a szó mondatbeli helye határozza meg. A *file* szó éppúgy lehet ige, mint köznév. Jelentése ugyanaz, csak a szerepe egyik esetben igei, a másikban főnévi. Persze lehet találni tisztán igei, és teljesen köznévi szavakat is, de nem ez a jellemző. Magyarban az igék és a névszók jobban elkülönülnek, de a névszók kategóriája szótani szempontból kevésbé különböznek. Különbségek vannak, de ezek elsősorban nem ragozási, szóalaktani, hanem mondattani különbségek.

A szokásos névszói kategóriák a következők: főnév, melléknév, számnév, és névmás. [EL], [PF] Ez utóbbi szerencsétlen osztály, mert a szintakszis szempontjából sokkal fontosabb, hogy a névmás főnévi, melléknévi, netán számnévi-e, mint az, hogy névmás. A főnév alosztályai a szokás szerint köznév és tulajdonnév, míg a számnevek három alosztálya tő-, sor- és törtszámnév. A mellékneveket nem szokták tovább osztani, legfeljebb megemlítik, hogy alap, közép vagy felsőfokú-e a szó, kifejezés.

2. A névszói szerkezetről

Az általánosan elfogadott kategóriák a szótani vizsgálatoknál használhatóak, de elégtelen a mondattanihoz. Én most elsősorban a névszói szerkezetek szempontjából vizsgálom a szófajokat.
Névszói szerkezetnek – esetünkben a magyar nyelvre vonatkozóan – nevezem azokat a nem igei szerkezeteket, melyeket kötött sorrenddel kell írni a magyarban. Ezek azok a szerkezetek, melyek részei az utolsót kivéve különböző alanyesetű névszóból, illetve alanyesetű jelzői szerkezetekből állnak, esetleg névelővel, kvantorral kezdve. (a melléknév jelentésének egy kiterjesztése a jelzői szerkezet több szóból álló szintagmákra).
Pl.: *mindhárom sárga csőrű pelyhes kiskacsával*

Az ilyen szerkezetekben a szórend kötött. A hagyományos szófaji beosztás alapján a szó végén egy köznév áll, ezt különböző jelző(s szerkezete)k előzik meg. Ezek előtt lehet egy mennyiségjelző, egy birtokos jelző, és legelöl a névelő és/vagy kvantor. Minden része opcionális, tehát nem kötelező, de (a névelőn kívül) legalább egy elemet kell tartalmaznia.

mindhárom pelyhes kiskacsával *mindhárom sárga csőrű pelyhessel*
mindhárom sárga csőrűvel *mindhárommal*

A *sárga csőrű* egy jelzői szerkezet az adott esetben. Kérdés, lehet-e a fenti kifejezésnek vele egyenértékű sorrendezése. Mi az, ami szintaktikus szabályokkal biztosítja a fenti sorrendet? Talán egyetlen más sorrend fogadható el, de azt sem érzem teljesen azonosnak: *mindhárom pelyhes sárga csőrű kiskacsával*
Másik példámban jobban látszik a kötöttség.

Kovács Katalin sikeres magyar sportolónő
tulajdonnév melléknév melléknév köznév
tulajdonnév melléknév köznév köznév

A hagyományos szófaji felosztás szerint a két melléknév közt nincs különbség szófaji felosztás szerint, ezért semmi sem indokolja, hogy ne cserélhetnénk fel. Amennyiben a szót főnévnek tekintjük, akkor viszont az utolsó két szó sorrendjét lehetne megváltoztatni. Mindkét megoldás elvetendő.

3. A névszók finomabb osztályozása – pontosabb névszói szerkezet

Ennek egy megoldását mutatja az 1988-ban készült kísérleti magyar mondatelemző [FN]. Ebben a munkában a következő új névszói szófaj szerepel: népcsoport, foglalkozás, mértékegység. A szófajok szemantikainak tűnnek, de nem tisztán azok. A lényeg, hogy a jelzők osztályozásánál a köznév és a különböző minősítésű jelzők közt sorrendi precedenciát lehet felállítani. Nem lehet tisztán főnév és melléknevekről beszélni. Ahelyett, hogy kétarcú névszókról beszélnénk, melyek a hagyományos beosztás szerint lehetnek pl. főnevek és melléknevek is, olyan kategóriát lehet nekik tulajdonítani, amelyek nem köznevek, nem is melléknevek, hanem a (prioritás alapján) a kettő közti szófaj. A jelenleg kísérleti beosztásom és azok közti prioritás a következő: ...<szelektor<mennyiség<tulajdonság<fajta<köznév

Magyarázat:
köznév a szokásos szófaj, ha nem teszem másik osztályba.
fajta: ezek többsége a köznév gyakori jelzői lehetnek. Személy esetén három alkategóriája van, a következő prioritással

népnév<vallás/pártállás<foglalkozás

Pl. *magyar református ács*
Állatoknál az **alfajt** jelentő jelző: *kardfogú tigris*
Élettelennél ilyen az **anyagnév**: *vörösréz névtábla*
Ez utóbbi két eset persze csak akkor érdekes, ha nem írják egybe az őt követő köznévvel
Tulajdonság: ezek az általános melléknevek, illetve jelzős szerkezetek. Jelzők lehetnek a szelektorban is.
Mennyiség: olyan részszerkezet, mely egy **számnév**ből és opcionális **mértékegység**ből áll esetleg egyszerűbb jelzővel díszítve. A számnéven itt csak tő- illetve tő- és törtszámnévből álló kifejezést értek, melyet most nem részletezek. A mértékegység tulajdonképpen köznév a hagyományos szófaji kategória szerint. Szerepe persze jól elkülöníthető az egyéb köznévtől.
A **szelektor** is több részre osztható: birtokos<pozíció=kiválasztó
Birtokos jelző a szokásos birtokost megjelölő részkifejezés, amelynek száma, személye az egész kifejezés birtokragjának számával és személyével megegyezik.
Pozíció vagy egy **sorszám**, vagy egy úgynevezett **pozícionáló melléknév**. A sorszám a szokásos, a pozicionáló melléknév pedig a melléknevek azon alosztálya, amely a középfok *bb*-je nélkül is kaphat *leg-* felsőfokot: *utolsó, szélső, hátsó...*
A **kiválasztó jelző** első pillanatra általános jelzőnek tűnik. Nem is lehet sokszor megkülönböztetni őket. A humán ellenőrzés, vajon egy jelző a tulajdonság, avagy a szelektor kategóriába tartozik a rákérdezés módszere. Ha a *milyen* kérdésre válaszol, akkor tulajdonságjelző, ha *melyik* kérdésre, akkor szelektor. Egyes melléknevek egyértelműen szelektorok. Ezek az *-ik* végű melléknevek, mint a sorszámnevek, vagy pl. *jobbik, másik.* Általában az összetett szerkezetű jelzők is ide tartoznak, de ennek eldöntéséhez további vizsgálatra van szükség: *tegnap vásárolt, első helyet elért...* A mai magyar nyelvben a szelektort tartalmazó névszói kifejezés névelővel vagy kvantorral kezdődik. Részletesebben a névszói szerkezetről lásd [FN].

4. A szavak osztályozásának módszere

A jelenlegi nyelvi adatbázisok hagyományos szófajokat rendelnek a szavakhoz. A felhasználás céljából elengedhetetlen az új kategóriába való sorolás. Az előző fejezet ötleteket is ad egyes szófajok kimerítő feltérképezésére. A fajta szófaj többsége főnévként és melléknévként is szerepel az adatbázisokban. Ezek kigyűjtése után emberi erővel elvégezhető a népnév, foglalkozás stb. beosztás.
Mértékegység erősen véges halmaznak tűnik: *méter, liter, kilogramm, hüvelyk...* Ez nem egészen van így. A gyakoribbak természetesen egy fizika-, kémiakönyvből összeszedhetők. Vannak azonban „alkalmi" mértékegységek is. Ezek azok a fizikai objektumot jelentő köznevek, melyek mennyiségek összevetésére alkalmasak: *három szekér szalma, egy csipet só.* Ezeknek a szavaknak az a szótani tulajdonságuk, hogy megkaphatják a *-nyi* melléknévképzőt: *szekérnyi, csipetnyi, maroknyi.* Ha nagy korpuszokból kigyűjtjük az ilyen szavakat, akkor a gyakoribbakat mindenképpen megtaláljuk.

A pozicionáló mellékneveket szintén könnyű kigyűjteni, hisz szintén jellegzetes a morfológiai tulajdonságuk.

5. Néhány szerkezeti példa

A kórházban sok ápolónő román.

Ez a mondat nem tekinthető egy névszói kifejezésnek, mert a népnév nem követheti a foglalkozásnevet, ezért nem hiányos mondat, hanem szabályos nominális mondat két független névszói kifejezéssel.

A szülő tegnap érkezett.

Ez sem lehet egy névszói kifejezés, mert a *tegnap érkezett* jelző összetett, ezért a *szülő* (mint foglalkozásszerű jelentéssel bíró szó) nem előzheti meg a *tegnap érkezett* jelzőt, de mint melléknév sem, mert mint halmozott szelektor, a két rész közé vesszőt kéne tenni. Természetesen szemantika nélkül így is marad „melléelemzés": az *érkezett* lehet ige és befejezett melléknévi igenév, sőt a *szülő* lehet a *tegnap* jelzője, vagy foglakozása. Ezen a módszer nem segít, de szemmel láthatóan csökkenti a többértelműségek számát.

6. Összefoglalás

A szófajok nyelvfüggő kategóriák. Jó osztályozás elősegíti a pontosabb szóalaktani és mondattani elemzést. A szófaji kategóriák kiválasztása modellalkotási probléma. A szófajokat úgy kell létrehozni, hogy egyrészt azok szóalaktani és mondattani szempontból relevánsak legyenek, másrészt algoritmikusan meghatározhatók legyenek. Segítségükkel a mondat szerkezete világosabb lesz, értsd ezen, kevesebb félreértelmezés jön ki az elemzés során. A jelenleg lefutott projektekben nem volt szükség alapos magyar mondatelemzésre, de a hamarosan beinduló magyarról való fordításnál elengedhetetlen pontosabb szerkezeti felbontás. Ez nem zárja ki más módszerek használatát, de mivel kellően általános, bizonyára segíteni fog.

Hivatkozások

1. [EL] Elekfi László: Magyar ragozási szótár. *MTA Nyelvtudományi Intézete*, Bp., 1994.
2. [FN] Farkas Ernő, Naszódi Mátyás: Magyar nyelvű mondatok elemzése természetes nyelvű interfész céljából. *SzTAKI kiadvány*, 1990 május,.
3. [KF] Kiefer Ferenc (szerk.): Strukturális magyar nyelvtan 3. Morfológia. Bp., *Akadémiai. Kiadó* 1994.
4. [PF] Papp Ferenc: A magyar főnév paradigmatikus rendszere Bp., 1975.

Az első nganaszan szóalaktani elemző

Novák Attila

MorphoLogic Kft. 1126 Budapest Orbánhegyi út 5.,
novak@morphologic.hu

Kivonat A cikk egy kihalás szélén álló kis északi szamojéd nyelv, a *nganaszan* szóalakjainak elemzésére hivatott programot mutat be, amely egy több uráli nyelvet felölelő projektum keretében készült el. Erre a nyelvre – annak rendkívül bonyolult fonológiája miatt – a projektum keretében egyébként használt *Humor* nyelvleíró formalizmus alkalmazása nagyon nehéznek bizonyult, ezért végül a Xerox cég *xfst* programjának felhasználásával készítettük el az elemzőt.

1. Bevezetés

Ebben a cikkben egy nganaszan nyelvű szóalaktani elemzőprogram kifejlesztéséről számolunk be. Erre a nyelvre eddig nem készült morfológiai elemzőprogram. Vállalkozásunk egy olyan projektum[1] részeként valósult meg, melynek célja elemzett korpuszok és egyéb elektronikus nyelvi erőforrások létrehozása néhány kisebb, az uráli nyelvcsaládba tartozó (tehát a magyarral valamilyen fokon rokon) nyelven.

Az Uráli nyelvcsalád északi szamojéd ágához tartozó nganaszan több szempontból is érdekesnek bizonyult a leírandó nyelvek közül. Egyrészt egy gyakorlatilag a kihalás szélén álló nyelvről van szó (beszélőinek száma már nem éri el az ötszázat, ezek túlnyomó része középkorú vagy idős, és az orosz kisebbségi politikának megfelelően nincs nganaszan nyelvű oktatás), ezért fontosnak tartottuk, hogy legalább a nyelv dokumentálásához hathatós segítséget nyújtsunk. Másrészt a nyelv morfológiája és különösen a fonológiája olyan bonyolult, hogy komoly kihívást jelentett a számítógépes formalizálása: elsőként a MorphoLogic Humor formalizmusát próbáltuk alkalmazni, de ebben a formalizmusban nem sikerült teljes leírást készítenünk a nyelvről. Végül a Xerox cég reguláris relációkalkuluson alapuló morfológiai fejlesztőrendszerének felhasználásával készítettük el az elemzőt.

[1] Komplex Uráli nyelvészeti adatbázis, NKFP 5/135/2001. A projektumban a Nyelvtudomány Intézet Finnugor Osztálya, különböző finnugor nyelvészeti tanszékek és a MorphoLogic Kft. vesz részt.

2. Első lépések a nganaszan morfológiai leírás létrehozására

Az Uráli Nyelvészeti projektumban eredeti terveink szerint a *MorphoLogic Kft. Humor* morfológiai elemzőprogramjának formalizmusát kezdtük el használni az egyes nyelvek morfológiájának leírására, pontosabban egy olyan nyelvészeti adatbázisleíró keretrendszert, amely az eredeti Humor formalizmusnál magasabb szintű, és ezért sokkal jobban karbantartható nyelvi leírás elkészítését teszi lehetővé, amiből automatikusan létrehozza a Humor morfológiai elemző által használt adatbázist (Novák (2003) [3]). A Humor formalizmusban a szavak morfémák egymáshoz illeszthető allomorfjainak jól formált sorozataiként épülnek fel. A szomszédos morfémák egymáshoz illeszthetőségének leírására a Humor jegyalapú formalizmust használ. A keretrendszert sikeresen alkalmaztuk különböző nyelvek leírására az Uráli nyelvleíró projektum keretében és azon kívül is.

A nganaszan nyelvről igen jó formális igényű leírást készítettek a projektumban részt vevő nyelvészkollégák (Wagner-Nagy (2002) [4]), és gépre vitték, majd az általuk használt latin betűs fonematikus átírásra konvertálták egy nganaszan-orosz szótár anyagát (Kost'erkina és mtsai. (2001) [2]). A szótár kb. 3650 tövet tartalmaz. Mindegyik tételhez kézzel beírták a megfelelő szófajmegjelölést is, amely az eredeti nyomtatott szótári anyagban nem szerepelt. Ilyen feltételek mellett azt reméltük, hogy viszonylag gyorsan és problémamentesen elkészülünk az elemzővel.

Az említett szótár alapján készülő tőtár gépre vitelével párhuzamosan közösen hozzákezdtünk a toldalékok formális leírásához. Első lépésként készült egy olyan toldaléklista, amelyben az egyes toldalékok mögöttes fonológiai alakja és kategóriacímkéje szerepelt, valamint az, hogy a toldalék melyik morfológiai tőalakhoz járul. A nganaszan morfológia leírásánál hasznosnak bizonyult az a modell, amely minden tő esetében három morfológiai tőváltozatot feltételez (ezek közül kettő vagy akár mindhárom is alakilag egybeeshet), és az egyes toldalékokat aszerint kategorizálja, hogy az első, második, vagy a harmadik tőváltozathoz kapcsolódnak-e. Néhány toldalék (pl. a latívusz esetrag) ingadozó viselkedést mutatnak: két különböző tőváltozathoz is kapcsolódhatnak. A mögöttes fonológiai leírás tartalmaz olyan magánhangzó-szimbólumokat, amelyek a nganaszan magánhangzó-harmónia szabályai szerint a tő harmonikus tulajdonságainak megfelelően váltakozó magánhangzókat jelölik. A képzők esetében ez az elsőként elkészült leírás még azt az információt is tartalmazta, hogy a képző milyen kategóriájú tőből milyen kategóriájút képez.

Következő lépésként a toldaléktárat a Humor elemzőhöz készített morfológiai-adatbázis-készítő keretrendszer által megkövetelt formájúra alakítottuk finomítva a leírás részletességét és hozzáadva a formalizmus által megkövetelt adatokat. Néhány egyértelműen szegmentálható (tisztán agglutinatív szerkezetű) komplex toldalékot szegmentáltunk (pl. az alany/tárgyeset + kettes szám + birtokos végződés alakú toldalékkomplexumokat). A toldalék által szelektált tőváltozatot (első, második vagy harmadik tő) bal oldali toldaléktulajdonságként fogalmaztuk meg. Bal oldali (a tőre tett) megszorítások elsősorban az igei végződéseknél szerepeltek: kizárólag perfektív, ill. kizárólag imperfektív igetövekhez járuló vég-

ződések, csak ágenses igék végződései, csak tranzitív igékhez járuló végződések stb.

A toldalékmorfémák sorrendjére vonatkozó morfotaktikai megszorításokat úgy írtuk le, hogy egyrészt a toldalékokat morfotaktikai osztályokba soroltuk: pl. alany/tárgyesetű birtokos végződések (NomPx), a többi esetben használt birtokos végződések (Px), oblikvuszi esetrag (OblCx), névszói predikatív végződés (NVx) stb. és leírtunk egy véges állapotú automatát, amelynek élein az egyes morfotaktikai osztályok címkéi szerepelnek, és azt írja le, hogy az egyes osztályokba tartozó morfémák milyen sorrendben követhetik egymást. A képzők esetében a megengedett sorrendek jelenlegi modellünkben egyszerűen abból következnek, hogy az adott képző milyen kategóriából milyen kategóriába képez.

3. A tő- és toldalékalternációk leírása, a tőtár rendhagyó lexikai jegyekkel való gazdagítása

A toldaléktár elkészítése után hozzákezdtünk a tőalternációkat leíró szabályok megfogalmazásához (a Humor keretrendszer ezek alapján a szabályok alapján állítja elő a morfématárakból az allomorfok adatbázisát). A névszói és az igei tövek különböző tőalternációs mintákat követnek. Ezeken belül is a magánhangzó- és a mássalhangzó-végű tövek jelentősen különböző viselkedést mutatnak. Bizonyos tővégi váltakozások csak a megfelelő lexikai jeggyel bíró tövek esetében fordulnak elő, hasonlóan pl. a magyar tőbelseji magánhangzó-rövidüléshez (ilyen váltakozás pl. a tővégi felső nyelvállású magánhangzók *a*-vá válása a névszóknál harmadik tőben), mások minden olyan tőnél jelentkeznek, amely a megfelelő alaki tulajdonságokkal rendelkezik, hasonlóan a magyar tővégi magánhangzónyúláshoz (ilyen pl. a tővégi *ja jai*-ra változása a névszóknál a harmadik tőben). Az előbbi kategóriába tartozó szavak mindegyikét meg kellett jelölni a tőtárban a megfelelő lexikai jeggyel.

4. A nganaszan morfofonológia bonyolultsága

A tővégi hangzók váltakozásait a Humor rendszer keretei között viszonylag egyszerűen le lehetett írni. A produktív fonológiai folyamatok közül a viszonylag lokális kontextusra érzékeny szabályokat (pl. degemináció) külön-külön meg tudtuk fogalmazni, azonban amikor a fokváltakozás jelenségét (a szótagkezdetben levő zárhangok szabályszerű váltakozását) is megpróbáltuk bevonni a leírt jelenségek körébe, kudarcot vallottunk. A probléma gyökere tulajdonképpen az a tény, hogy a Humor elemző az elemzendő szót mindig morfémák allomorfjainak sorozataként látja, és elemzés közben minden morfhatáron azt ellenőrzi, hogy a az előző és a következő morf tulajdonságai kölcsönösen kielégítik-e egymásnak a másikkal szemben támasztott megszorításait. Ez a modell eddig minden nyelv esetében jól működött, és általában nem okozott problémát ezeknek a morfok közötti megszorításoknak a megfogalmazása.

A nganaszan fokváltakozás azonban egyáltalán nem függ a szó morfológiai szerkezetétől: kizárólag a szótagszerkezet játszik szerepet benne. A szótaghatárok

és a morfémahatárok viszont általában semmilyen kapcsolatban sincsenek egymással, rövid (1 szegmentumból álló) toldalékok esetében (van ilyen toldalék a nganaszanban) még egymással nem szomszédos morfémák is közös szótagba kerülhetnek. Ez a tény súlyosbítva azzal a körülménnyel, hogy a fokváltakozásban nemcsak az adott, illetve a megelőző szótag zártsága és az előző szótagban levő magánhangzó hosszúsága, hanem még az is szerepet játszik, hogy az adott szótag páros vagy páratlan sorszámú a szón belül, illetve hogy mindez az összes többi váltakozással kombinálódik (magánhangzó-harmónia, degemináció, a legkülönbözőbb tő- és toldalékalternációk és hasonulások; ezek együtt egy egyszótagú toldaléknál könnyen tizennégy különböző allomorfot eredményeznek) oda vezetett, hogy képtelenek voltunk a fokváltakozást (illetve a nganaszan morfofonológia egészét) felszíni allomorf-szomszédsági megszorítások együtteseként leírni, az allomorfokat és a közöttük fennálló megszorításokat előállító szabályegyüttest megfogalmazni.

Ízelítőül álljanak itt egy igei toldalék (az elbeszélő mód alanyi és tárgyas ragozásban használt formájának) allomorfjai. A morféma absztrakt lexikai alakja: hA_2nhV, allomorfjai (12 van, és ez még nem is a legbonyolultabb eset): *banghu, bjanghy, bambu, bjamby, bahu, bjahy, hwanghu, hjanghy, hwambu, hjamby, hwahu, hjahy.* Az allomorfok szabályszerűen állnak elő az alábbi fonológiai folyamatok eredményeképp:

Az A_2 harmonikus magánhangzó *a*-ként vagy *ja*-ként jelenik meg a tőhöz a kerekítési harmónia szabályai szerint illeszkedve, ráadásul az *a h* után szabályszerűen *wa*-vá diftongizálódik. (Hogy egy tő melyik kerekítési harmóniai osztályba tartozik, az teljesen megjósolhatatlan lexikai jegye, a tövek egy része ingadozó viselkedést mutat.) A V harmonikus magánhangzó *u*-ként, *y*-ként, *ü*-ként vagy *i*-ként jelenik meg a kerekítési és az elölségi harmónia szabályainak megfelelően (ebben a toldalékban csak *u* és *y* lehetséges, mert az előző szótagban mindenképpen hátul képzett magánhangzó (*a/ja/wa*) van). A *h* fonéma erős fokban *h*, gyenge fokban *b*. Az *nh* kapcsolat erős fokban, vagy ha az ún. nunnáció fellép *ngh*, ritmikai gyege fokban *h*, szillabikus gyenge fokban *mb* (a nazális képzési hely szerint illeszkedik, ritmikai gyege fokban viszont eltűnik, hacsak az előző mássalhangzó nem nazális: akkor ugyanis megmarad, ez a nunnáció). Erős fokban áll egy szótagkezdő obstruens, ha nem nazális kóda (mássalhangzó) előzi meg, vagy ha a szón belül páros sorszámú nyílt szótagban van. Egyébként ritmikai gyenge fokban áll, ha hosszú magánhangzó előzi meg, vagy páratlan szótagban áll, és szillabikus gyenge fokban áll, ha páros zárt szótagban áll.

A fokváltakozás annyira produktív folyamata a nganaszan fonológiának, hogy formális leírását semmiképpen sem kerülhetjük el, ha működő elemzőt akarunk készíteni. Úgy tűnt azonban, hogy bár a Humor formalizmusa nem eleve alkalmatlan ennek a nyelvnek a leírására, de a gyakorlatban legalábbis a keretrendszer szabályformalizmusának felhasználásával a leírás elkészítése túl nehéz feladatnak bizonyult.

5. Áttérés egy új formalizmusra

2003 júniusában azonban megjelent egy könyv (Beesley-Karttunen (2003) [1]), amelyhez mellékletként adtak egy CD-t, amelyen a *Xerox* cég véges állapotú automata alapú kétszintű morfológiai elemző készítő programcsomagjának nem üzleti célokra szabadon felhasználható verzióját szabadon hozzáférhetővé tették. A programot kutatóintézetek kutatási célra korábban is licenszelhették, de ez olyan hosszadalmas jogi procedúrával járt, és annyi feltételnek kellett eleget tenni, hogy azelőtt szinte elképzelhetetlen volt, hogy hozzájussunk ennek a projektnek a keretében.

Most azonban rendelkezésünkre állt, és úgy döntöttünk, hogy a nganaszanról készített leírásunkat átalakítjuk, illetve újraírjuk a Xerox *lexc* (Lexicon Compiler), illetve *xfst* (Xerox Finite-State Tool) programjai által megkövetelt formában. A *lexc* programmal morfématárakat lehet definiálni folytatási osztályok megadásával, az *xfst* pedig a generatív fonológusok által megszokott kontextusfüggő újraírószabály-formalizmussal leírt szekvenciális fonológiai szabályegyüttesek megadását teszi lehetővé, és kiszámítja az egyes szabályok egymással illetve a lexikonnal való komponálásával előálló teljes morfofonológiai leírást egyetlen kétszintű véges állapotú fordítóautomata formájában, amit elemzésre és generálásra egyaránt lehet használni.

Az *xfst* formalizmusában nem jelentett többé áthághatatlan akadályt a fokváltakozás leírása, mert a program által megvalósított kalkulus lehetővé teszi, hogy az újraíró szabályok környezetmegadásánál az irreleváns szimbólumokat (pl. a morfémahatárokat) figyelmen kívül hagyjuk, ugyanakkor nem jelent problémát a nem szomszédos morfémákra átnyúló környezetek figyelembe vétele sem. Mivel a program az egyes szabályok által létrehozott köztes szinteket a kompozíció révén automatikusan eliminálja, semmilyen hatékonysági problémához nem vezet elemzés és generálás közben a leírás elkészítésekor bevezetett nagy számú közbülső leírási szint sem.

A fokváltakozást az *xfst* formalizmusában úgy írtuk le, hogy definiáltunk egy szabályegyüttest, ami a szótaghatárokon explicit határszimbólumokat illeszt be (a páros és a páratlan szótagok között más-más szimbólumot), az előző szótag zártsága, a benne szereplő magánhangzó hosszúsága, valamint az adott szótag zártsága és páros vagy páratlan volta alapján erős vagy ritmikai/szillabikus gyenge fokúként jelöli meg az egyes szótagokat, majd a szótagkezdetben levő mássalhangzót (illetve a szótaghatáron levő nazális-zárhang kapcsolatokat) pedig a szótag fokának megfelelően megváltoztatja, végül a szótaghatár és fokszimbólumokat kitörli. A szabályrendszer kezeli a nganaszan kivételes szótagolási jelenségeit is: a gégezárhang akkor is zárja a szótagot, ha nem követi másik mássalhangzó (a *V'V* sorozat szótagolása: *V'.V*), a *bt* hangkapcsolat *b*-je viszont nem zárja a szótagot (*V.btV*).

A konkrét szabályegyüttes alább látható:

```
#a dot after every syllable that is followed by an onset
[[C* V C*]/NSeg @-> ... "." || _ [C V]/NSeg ]
#a dot before syllables without an onset
.o.[ V @-> "." ... || V/NSeg _ ]
#resyllabify ' from onset to coda: insert syllable boundary after '
.o.[' -> ... "." || "."/NSeg _ ]
#delete syllable boundary before '
.o."." -> 0 || _ [' "."]/NSeg ]
#resyllabify b from coda to onset if followed by t:
#insert syllable boundary before b
.o.[b -> "." ... || _ ["." t]/NSeg]
#delete syllable boundary after b
.o."." -> 0 || ["." b]/NSeg _ t/NSeg]
#strong grade after non-nasal codas and m codas not followed by b
.o."." -> ... "^S" ||
[C-[n|m|n1|ng|N|M|N1|NG]]/NSeg _ , [m|M]/NSeg _ [Seg-[b|B]]/NSeg]
#rhytmical weak grade after long vowels
.o."." -> ... "^W1" || [V V]/NSeg _ ]
#change every second dot to a comma:
#. = even syllable
#, = odd syllable
.o."." -> "," \/ "." ~$["."|","] _ ]
#rhytmical weak grade in odd syllables not yet marked as strong
.o.[","  -> ... "^W1" || _ NGrd]
#syllabic weak grade in even closed syllables not yet marked as weak
.o."." -> ... "^W2" || _ [NGrd ?* & [C* V C]/\[Seg|"."|","]]]
#strong grade in other even syllables (codaless ones)
.o."." -> ... "^S" || _ NGrd]
#gradation
#rhythmical weak grade of obstruents
"^W1" h -> b, "^W1" t -> q, "^W1" k -> g, "^W1" s -> d1,
"^W1" s1 -> d1 || NNas /NSeg _ ,,
#rhythmical weak grade of nasal+obstruent clusters
Nas -> ^N || _ [["."|","] "^W1" [h|k|t|s|s1]]/NSeg,,
#syllabic weak grade
"^W2" h -> b, "^W2" t -> q, "^W2" k -> g,
"^W2" s -> d1, "^W2" s1 -> d1
#remove syllable boundaries
.o."^W1"|"^W2"|"^S"|"."|","] -> 0
```

A szabályrendszer ezek mellett rengeteg más szabályt tartalmaz, egyrészt produktív automatikus fonológiai szabályokat (pl. a nazálisok hely szerinti hasonulása a következő obstruenshez, degemináció, magánhangzó-harmónia, nunnáció, palatalizáció stb.), másrészt a morfológiailag, ill. lexikálisan megszorított,

szűkebb körben működő tő- és toldalékalternációkat is ilyen szabályokkal lehetett az új rendszerben leírni.

6. A morfématárak konverziója

Természetesen az új formalizmus nemcsak az allomorfiák leírásában különbözik gyökeresen a korábban használttól, hanem a morfématárak és a morfotaktika megadásának módjában is. Gondoskodnunk kellett tehát egy olyan konverterről, amely meglévő morfématárainkat a megfelelő formátumra konvertálja. A Humor leírás alapjául szolgáló jegyalapú megszorításokat alkalmazó formalizmus a morfológiai megszorítások (pl. a toldalékok tőszelekciója) leírásában igen jól használhatónak bizonyult, bár a nagyon komplex felszíni fonológia leírására – mint láttuk – nem bizonyult a leghatékonyabb eszköznek. Szerencsére a Xerox elemző formalizmusa is tartalmaz jegy-érték megszorítások leírására alkalmas eszközt (Flag Diacritics), ezért ezeket a megszorításokat hasonló elven meg lehet fogalmazni, mint egy Humor elemző készítésekor.

A lexikon leírására használatos *lexc* program által használt formalizmusban a lexikon morfémák leírását tartalmazó allexikonok sorozatából áll, minden egyes morfémához meg kell adni egy folytatási osztályt, ami vagy egyszerűen annak az allexikonnak a neve, amelynek összes tagja követheti az adott morfémát, vagy a szó végét jelölő szimbólum.

Az alábbi példa a Humor keretrendszer által használt toldaléklistából az elbeszélő mód alanyi és tárgyas ragozásban, illetve a visszaható ragozásban használt formájának leírását mutatja.

```
#mode suffixes
#tag    phon      lp    mcat     comment
(...)
Narr    HA2NHU    S1    VTM      narrative subj/obj
Narr    HA2NHA1   S1    VTMR     narrative refl.
```

Ugyanezek a toldalékok a *lexc* program által használt formára konvertálva így néznek ki:

```
LEXICON infl_V
(...)
@U.S.1@@C.S@h^A2nh^V[Narr]      infl_VTM_r;
@U.S.1@@C.S@h^A2nh^A1[Narr]     infl_VTMR_r;
```

Az `@U.S.1@` szimbólum azt jelöli, hogy az adott toldalék az első morfológiai tőalakhoz járul (az `@U.S.1@` jelentése: 'unifikáld az S tulajdonság aktuális értékét az 1-es értékkel'). A `@C.S@` szimbólum a semleges értékre állítja (törli) az S tulajdonság értékét. A képzők, mint tövek allomorfjainak előállításáról, és a tőtípust azonosító jegyek kitöltéséről a szabályos tőalternációkat leíró szabályok gondoskodnak.

Az elkészült elemzőt latin betűs fonológiai átírással lejegyzett nganaszan nyelvű szövegek morfológiai elemzésére fogjuk felhasználni. Az elemzett szövegeket a projektum honlapján tesszük majd közzé.

Hivatkozások

1. Kenneth R. Beesley and Lauri Karttunen. *Finite State Morphology.* CSLI Publications, Ventura Hall, 2003.
2. N. T. Kost'erkina, A. Č. Momd'e, and T. Ju. Ždanova. *Slovar' nganasansko-russkij i russko-nganasanskij.* Prosvesčen'ije, Sankt-Pet'erburg, 2001.
3. Novák Attila. Milyen a jó humor? In *Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2003)*, pp. 138–145, Szegedi Tudományegyetem, 2003.
4. Wagner-Nagy Beáta, (szerk.) *Chrestomathia Nganasanica.* SZTE Finnugor Tanszék – MTA Nyelvtudományi Intézet, Szeged – Budapest, 2002.

Javaslat az etimológiai minősítés egységesítésére

Sass Bálint

MTA Nyelvtudományi Intézet, Budapest
joker@nytud.hu

Kivonat Elektronikus nyelvi erőforrások esetén az egységes formátum mindig egyszerűbb feldolgozhatóságot és könnyebb felhasználhatóságot jelent. Dolgozatomban az összegyűjtött követelmények alapján kialakítom az etimológiai minősítés egyfajta egységes, a korszerű, XML alapú lexikai adatbázisokban alkalmazható modelljét. A rendszer működését példákon keresztül mutatom be.
Kulcsszavak: etimológia, egységes etimológiai minősítés, DTD

1. Előzmények

A Webster szótár 1828-as kiadása óta terjedt el, hogy a szótárak a szócikk alapvető részének tekintik az etimológiai minősítést [2], mely „azt van hivatva megjelölni, hogy a kérdéses szó eredete sommásan miként határozható meg." [1]

Benkő Loránd a téma szempontjából nagy jelentőségű cikkében hangsúlyozza az egységes etimológiai szemlélet és ennek folyományaként az egységes etimológiai minősítési rendszer szükségességét. „Szabatos etimológiai minősítések hiányában egyáltalán nem lehetne statisztikát készíteni egy-egy nyelv szókészletének származásbeli kategóriáiról." Fontos szerepet tulajdonít a különféle szóalkotási módok számbavételének. Elvként fogalmazza meg, hogy egy nyelvi elem történetében azokat a pontokat kell megragadni, amikor az elem újnak vehető minőségbe lép, illetve hogy a közvetlen etimológiai előzményhez kell viszonyítanunk, azaz a lépések közül számunkra mindig a legutolsó, az időben hozzánk legközelebbi a legfontosabb. Megjelenik az a sokak által fontosnak tartott követelmény is, hogy azonos jelenségeket konzekvensen azonos, különbözőeket különböző minősítéssel kell ellátni [1].

A különféle szótárak etimológiai minősítési rendszere meglehetősen eltérő. Tapasztalatok szerint az etimológiai leírás szinte bármit tartalmazhat egy egyszavas meghatározástól egészen a szó történetéről szóló terjedelmes esszéig [2], ezenkívül a szótárakban az etimológiai információ jelentős része rejtetten jelenik meg.

Dolgozatom célja egy a fenti kívánalmaknak megfelelő és az említett hátrányokat kiküszöbölő egységes etimológiai minősítési rendszer kidolgozása absztrakt modell, illetve konkrét XML DTD formájában. Ezutóbbi teszi lehetővé, hogy a modellt lexikai adatbázisokban alkalmazzák.

Napjainkig jellemző, hogy a lexikai adatbázisokban az etimológiai minősítést egyszerű szövegként kezelik. Lemondanak a nyelvi elemek közötti kapcsolatok

ábrázolásáról [7]. Olyan modell, ami minősítés a belső szerkezetét megragadná tudomásom szerint még nem készült. Az általam ajánlott modell beilleszthető bármelyik formátumba az egyszerű `<etym>` tag helyére.

A kiinduló anyagot az ÉKSz-ben [6] található etimológiai minősítések adták. Jelen dolgozatban a modell ennek megfelelő részletességű kidolgozását tűztem ki célul, vállalva, hogy ezzel leegyszerűsítem a problémát.

2. Modell

Nézzük azokat a tulajdonságokat, aminek a modell meg kell, hogy feleljen.

- Legyen *egyetemes*: képes legyen magába foglalni minden lehetséges etimológiai megoldást [5].
- Legyen *rugalmas*: majdnem minden részlet megadása opcionális legyen.
- Legyen *érzékeny*: ragadja meg a nyelvi elem összes minőségi változását.
- Legyen *többrétű*: engedjen meg alternatívákat és legyen alkalmas eszköze valószínűségi kategóriák kifejezésére.
- Legyen *explicit*: tegye könnyen hozzáférhetővé az információt.
- Legyen *konzekvens*: azonos jelenséget azonos módon, különbözőt különböző módon kezeljen.

Az egyetemesség kívánalmába beletartozik, hogy a nyelvi elemekhez időpontot is rögzíteni lehessen. A köznévi és a tulajdonnévi eredet jól elkülönül, ezt érdemes jelölni. A nyelvi elemekhez általában vagy a forrásnyelvet adják meg vagy a konkrét szót, amiből származik. Hasznos, ha opcionálisan mindkettőt meg lehet adni.

Alább a modell két formájának közös leírása következik, zárójelben mindig utalok a DTD vonatkozó kulcsszavára.

A modell lényegi ötlete, hogy az etimológiai minősítést *elem*ek (`elem`) és rajtuk értelmezett *művelet*ek (`op`) rendszerének fogja fel. Az elemek egyszerűen fogalmazva szavak, a műveletek pedig a különféle szóképzési módok. Szavakból kiindulva a műveletek segítségével új szavak képződnek, míg végül elérkezünk a minősítendő szóhoz. A *lépés* (`step`) összefoglalja a műveletet az operandusként hozzátartozó elemekkel, az etimológiai minősítést így lépések sorozatának lehet tekinteni. Szokás szerint a lépések a jelenből haladnak a múlt felé, a minősítés legfontosabb részét az első lépés adja. Az etimológiai minősítés tehát egy fastruktúra alakját veszi fel, ahol a fa gyökerében a minősítendő szó foglal helyet, a csúcsok az elemeket, az élek, élcsoportok a műveleteket jelentik, adott csúcsba vezető élcsoport a kiinduló csúcsokkal együtt pedig egy lépést alkot.

Igyekeztem számbavenni az összes lehetséges műveletet [4]. Ezeket az alábbi táblázat foglalja össze, az ajánlott operandusszámmal, illetve a DTD-ben szereplő rövidítéssel együtt.

leszármazás	1	desc
átvétel	1	loan
nemzetközi szó és vándorszó	1	internat
tükörfordítás	1	calque
félig tükörfordítás	1	lc
származékszó	1	deriv
betűszó	2≤	acr
szóösszevonás, szóösszerántás	2≤	contr
szóösszetétel	2≤	comp
szóvegyülés	2≤	mix
hangutánzó, hangfestő szó	0	onom
gyermeknyelvi szó	0	child
mesterséges szóalkotás	0	artif
állathívogató szó, indulatszó	0	interj
egyéb kapcsolat	1≤	rel

Nulla az operandusszám, ahol nem adható meg elem, amiből a szó kialakult, egy ott, ahol egy szóból alakul ki egy másik, és kettő vagy több a „szóösszerakás" különféle fajtáinál. Szóösszetételnél legalább az egyik tagnak teljes szónak kell lennie, ha mindkét rész csonka, akkor szóösszevonás a művelet, a betűszóknál a tagokból csak egy-egy betű kerül át, szóvegyülésnél a betűk sorrendje is változhat.

Az elemeknél meg lehet adni a *nyelv*et (lang), a *szóalak*ot (orth), a *jelentés*t (def), a *magyarázat*ot (gloss), az *átadott alak*ot (pass), az *időpont*ot (time) és hogy tulajdonnévről vagy köznévről van szó (proper). Közneveknél általában a jelentés, tulajdonneveknél a magyarázat használandó. Az átadott alak arra szolgál, hogy például szóösszevonás esetén megadja az elem azon részletét, mely a műveletben részt vesz.

Az alternatívák lehetőségét műveletként volt a legkényelmesebb bevezetni (alt). Operandusszáma legalább kettő, eredménye pedig mindig üres elem. A valószínűségeket elem és lépés esetén is megadhatjuk (p). Öt valószínűségi osztályt hoztam létre, a *kizárt*tól (impossible) a *biztos*ig (sure). Alapértelmezett az utóbbi, és a kérdéses részletek esetén általában a középső *kérdéses* (quest) használandó.

A modell DTD formában a következőképpen néz ki.

```
<!ELEMENT  etym   (step+) >
<!ELEMENT  step   (elem*) >
<!ATTLIST  step
           op     (desc|loan|internat|calque|lc|
                  deriv|acr|contr|comp|mix|
                  onom|child|artif|interj|rel|
                  alt)                                  #REQUIRED
           id     ID                                    #IMPLIED
           p      (impossible|perhaps|quest|probable|sure) "sure" >
<!ELEMENT  elem   (lang?,orth?,def?,gloss?,pass?,time?) >
```

```
<!ATTLIST  elem
           proper (y|n)                                        "n"
           stepid IDREF                                        #IMPLIED
           p      (impossible|perhaps|quest|probable|sure)     "sure" >
<!ELEMENT  lang   (#PCDATA) >
<!ELEMENT  orth   (#PCDATA) >
<!ELEMENT  def    (#PCDATA) >
<!ELEMENT  gloss  (#PCDATA) >
<!ELEMENT  pass   (#PCDATA) >
<!ELEMENT  time   (#PCDATA) >
```

3. Példák

Végül nézzük meg néhány példán, hogy hogyan működik a gyakorlatban a rendszer. Három példa az ÉKSz-ből [6], a negyedik pedig egy angol szótárból való. Először mindig a címszót és az etimológiai minősítést idézem, majd rövid magyarázattal és a modell szerinti XML alak következik.

1. **hordár** [←*hord*]
Itt egyetlen lépésből áll a minősítés: egy művelet szükséges egy elemmel. A művelet a származtatás, az elem pedig a *hord* szó, melynek jelen esetben csak a külső alakja van megadva. Az XML forma így a következő lesz:

```
<etym>
 <step op="deriv">
  <elem>
   <orth>hord</orth>
  </elem>
 </step>
</etym>
```

2. **tegnap** [*teged* 'minap' (↔*té*)+*nap*]
Két lépést, műveletet különíthetünk el: egyrészt egy szóösszetétel szerepel, mely két elemet kapcsol össze, másrészt az előtag kapcsolatban áll egy további elemmel. Látjuk, hogy a *teged* szónak csak egy része kerül át a címszóba, ezt a `pass` tagban adhatjuk meg pontosan. Ha egy elem egy lépés (művelet) eredménye, akkor szükséges azonosítani, hogy melyik lépésé. Erre szolgál az `elem` tag `stepid` attribútuma, mely egy `step` tag `id` attribútumára hivatkozik. Az első lépés eredménye maga a címszó, az összes többi lépés eredményét viszont elemként használjuk fel, így ezeknek mindig meg kell adni az `id` attribútumát.

```
<etym>
 <step op="comp">
  <elem stepid="id0-es1">
   <orth>teged</orth>
   <def>minap></def>
   <pass>teg</pass>
  </elem>
  <elem>
   <orth>nap</orth>
  </elem>
 </step>
 <step op="rel" id="id0-es1">
  <elem>
   <orth>té</orth>
```

```
  </elem>
 </step>
</etym>
```

3. **fennhéjáz** [*fenn*+←?*héj* 'szemhéj' v. *héja*]
Ebben a példában megjelenik a kérdésesség és a vagylagosság. A kérdésesség itt az `elem` tag `p` attribútumában adható meg. A példában a kérdőjel vélhetően a *héj* és a *héja* szavakra is vonatkozik, az XML formából ez egyértelműen kiderül. A vagylagosság (`alt`) műveletének eredménye üres elem. Az XML formában a származékszó (jele itt: ←) művelet kapcsán még egy lépés megjelenik, ennek az eredménye is egy üres elem lesz, ami az eredeti formában az összeadásjel és a nyíl között kapna helyet.

```
<etym>
 <step op="comp">
  <elem>
   <orth>fenn</orth>
  </elem>
  <elem stepid="id1-es1"/>
 </step>
 <step op="deriv" id="id1-es1">
  <elem stepid="id1-es2"/>
 </step>
 <step op="alt" id="id1-es2">
  <elem p="quest">
   <orth>héj</orth>
   <def>szemhéj</def>
  </elem>
  <elem p="quest">
   <orth>héja</orth>
  </elem>
 </step>
</etym>
```

4. **polyp** [F f. L f. Gk (*pous* foot)]
A modell egyetemességének (ld. 204. o.) illusztrálására az utolsó példát a láthatóan más formátumot követő Pocket English Dictionaryból vettem [3], Ebben a minősítésben három művelet rejlik: a szót az angol a franciából, az a latinból, az a görögből vette át (az átvétel jele itt az 'f.'). Zárójelben az eredeti görög szó van megadva jelentéssel együtt.

```
<etym>
 <step op="loan">
  <elem stepid="id2-es1">
   <lang>F</lang>
  </elem>
 </step>
 <step op="loan" id="id2-es1">
  <elem stepid="id2-es2">
   <lang>L</lang>
  </elem>
 </step>
 <step op="loan" id="id2-es2">
  <elem>
   <lang>Gk</lang>
   <orth>pous</orth>
   <def>foot</def>
  </elem>
 </step>
</etym>
```

Hivatkozások

1. Benkő, L.: Az etimológiai minősítés a szótárszerkesztésben. Magyar Nyelv, XC. évf (1994), 4. szám, pp. 385-392.
2. Drysdale, P. D.: Etymological Information in the General Monolingual Dictionary. In Dictionaries – An International Encyclopedia of Lexicography (szerk.: Hausman, F. J., Reichman, O., Wiegand, H. E., Zgusta, L.) pp. 525-530.
3. Fowler, F. H.: The Pocket Oxford Dictionary of English. 7th ed. Clarendon Press, Oxford, 1984.
4. Keszler, B. (szerk.): Magyar grammatika. Nemzeti Tankönyvkiadó, Budapest, 2000.
5. Kiss, L.: Kísérletek etimológiai képletek felállítására. Magyar Nyelv, LX. évf (1964), 4. szám, pp. 314-321.
6. Pusztai, F. (szerk.): Magyar Értelmező Kéziszótár. Akadémiai Kiadó, 2003.
7. Sperberg-McQueen, C., Burnard. L. (szerk.): The XML Version of the TEI Guidelines. Print Dictionaries. The TEI Consortium, 2002. `http://www.tei-c.org/P4X/DI.html`

VI. Különböző szövegtípusok elemzése

A szavak véletlenszerű megjelenésén alapuló modellek és az irodalmi művek közötti eltérések magyarázata

Csernoch Mária

Angol-Amerikai Intézet 4010 Debrecen, Egyetem tér 1.
mcsernoch@hotmail.com

Abstract. Munkánkban arra vállalkoztunk, hogy kiderítsük, mi okozza egy természetes nyelvi szövegnek egy dinamikus lexikai statisztikai modelltől való eltérését. A feladat megoldásához egy saját fejlesztésű programot használtunk. A program segítségével meghatároztuk, majd ábrázoltuk a kiválasztott irodalmi művekben az újonnan megjelenő szóalakokat és vizsgáltuk azokat a pontokat, amelyek szignifikáns eltérést mutatnak a modellhez képest. Vizsgáltuk, hogy a szintaktikai szabályok befolyásolják-e, és ha igen mennyiben, a szóalakok megjelenését, illetve származhatnak-e más forrásból az eredeti és a mesterséges szöveg közötti eltérések. Az eredeti művet összehasonlítva a modellel, majd a mű fordításaival azt tapasztaltuk, hogy az eltérések nem szintaktikai, illetve szemantikai, hanem sokkal inkább szöveg szinten jelentek meg és okoztak látványos növekedést az újonnan bevezetésre kerülő szóalakok számában.

Bevezetés

Mindannyiunk által ismert és elfogadott tény, hogy irodalmi művekben a szavak nem egymástól függetlenül követik egymást. Ezzel látszólagos ellentmondásban a korábban megépített akár statikus, akár dinamikus modellek valamilyen szinten mind feltételezték a szavak egymástól független megjelenését az irodalmi művekben. Ez a feltételezés egy nyilvánvaló leegyszerűsítése a problémának, aminek következtében a felhasznált módszertől függően különböző mértékű, esetleg nagyságrendű eltérések tapasztalhatóak az eredeti mű és a modell között, valamint az eredeti mű tulajdonságait leírni szándékozó konstansok és a mért eredmények között (Baayen, 1993, 1996, 2001; Hoover, 2003). Ugyanakkor napjainkra az is elfogadott, hogy a szavak függetlenségét feltételező modellek segítségével a szöveg bizonyos tulajdonságait leírni képes formulákat sikerült találni.

Vizsgálva a szavak nem-véletlenszerűségének forrásait azonban azt tapasztalták (Baayen, 1996, 2001), hogy bár a mondaton belüli kötöttségek a legnyilvánvalóbbak, mégsem ezek a legfőbb forrásai a különböző szóalakok nem-véletlenszerű megjelenésének. Sokkal inkább meghatározónak bizonyult az, hogy milyen szerkezetű a mű.

Ezt bizonyítandó Baayen (Baayen, 1996, 2001) azt a módszert használta, hogy egy olyan mesterséges szöveget állított elő, amelyben véletlenszerűen összekeverte a szöveg mondatait meghagyva azonban a szavak mondaton belüli sorrendjét. Azt tapasztalta, hogy az így előállított mesterséges szöveg és az általa használt modell között

látványosan csökkentek, esetenként eltűntek azok az eltérések, amelyek az eredeti szöveg szóalakjainak száma és a modell által számolt szóalakok között még jelen voltak.

Vizsgálatainkban annak az állításnak a bizonyítására, hogy a véletlen szóhasználattól csak szöveg szinten térnek el az írók egy olyan módszert használtunk, ahol nem volt szükség mesterséges szöveg előállítására, hanem az eredeti művet hasonlítottuk össze az általunk használt dinamikus modellel (Csernoch, 2003; Csernoch és Hunyadi, 2003).

Módszerek

A szövegek elemzéséhez készítettünk egy programot (Csernoch és Hunyadi, 2003; Csernoch, 2004), amely megszámolta és tárolta a szövegben megjelenő szavakat. A tárolt adatok alapján, többek között, a program meghatározta az újonnan megjelenő szóalakok számát, az egyes szóalakok gyakoriságát, az egyszer előforduló szavak számát, stb. További vizsgálatainkhoz megszámoltuk, hogy száz-szövegszó-hosszúságú intervallumokban (blokkokban) hány új szóalak (y_i, $i = 1, \ldots, n$, ahol n a blokkok száma) jelenik meg az előzőekhez képest és az így kapott értékeket ábrázoltuk. A függvény monoton csökkenő tendenciáját megtörő kiugrások (1-3. ábra; pontok) a szövegben jelenlévő trendek és szezonalitások következményei. A trendek jelenlétére utaló kiugrásokat elsődleges, míg a szezonalitások következtében megjelenőket másodlagos kiugrásoknak nevezzük (1. ábra). Szezonalitások alatt értjük azokat az eseményeket, amelyek nem logikus következményei az előzményeknek és ennek következtében bevezetésükhöz kiugróan magas számú új szóalakra van szükség. A program az egyes blokkokban újonnan megjelenő szóalakok számának meghatározása (y_i) után elvégzi a függvény ábrázolását. A grafikonról az esetek többségében jól leolvasható, hogy melyek azok a pontok, ahol ezek a rendkívüli események bekövetkeznek, de a grafikon alapján nehéz megmondani, hogy mely változások tekinthetők szignifikánsnak. További feldolgozásra volt szükség tehát annak eldöntésére, hogy az újonnan megjelenő szavakat leíró görbe mely csúcsai tekinthetők elsődleges, illetve másodlagos kiugrásnak.

Elsőként a mért adatok alapján kapott görbe simítását kellet elvégezni, az így kapott értékek (y'_i) az f' simított görbe függvényértékei. A száz szövegszó hosszúságú blokkok ugyanis kellően rövidek ahhoz, hogy visszaadják a szöveg finomabb változásait is, de éppen e miatt a jelentéktelen változásokra is érzékenyek. Amennyiben a szövegben bekövetkezett változás jelentéktelen, csak abban az egy blokkban érezteti hatását, úgy az a simítás során eltűnik, ugyanakkor a jelentős változások a simítás után is megjelennek a görbén (1/A-3/A. ábra; folytonos vonal).

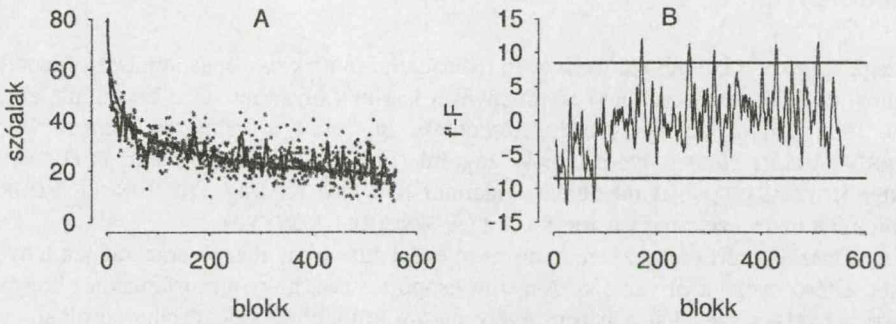


Fig. 2. Kertész Imre SORSTALANSÁG című művének elemzése. Az újonnan megjelenő szóalakok száma, a simított görbe és száz modell átlaga (A). A simított görbe és az átlag függvény közötti eltérés (B).

Ezt a simított görbét hasonlítottuk a modell által előállított mesterséges szöveg szóalakjait leíró görbék sorozatához (fp_k, k = 1, ..., 100), ahol yp_{ki} jelöli a k. függvény i. blokkjában megjelenő szóalakok számát. A modell alapján előállítottunk száz mesterséges szöveget, megszámoltuk ezen szövegekben az újonnan megjelenő szavak számát a száz szövegszó hosszúságú blokkokban és vettük az így kapott függvények átlagát (F) (1/A-3/A. ábra) (Ashby, 1972).

$$F = \frac{\sum_{k=1}^{100} yp_{ki}}{100} \tag{1}$$

A következő lépésben vettük a simított függvény és az átlag függvény különbségét (Δf, melynek függvényértékei Δy_i)

$$\Delta f = f' - F, \tag{2}$$

$$\Delta y_i = f'_i - F_i,\ i = 1, \ldots, n, \tag{3}$$

majd a különbségek átlagát (M) és szórását (σ) (Hajtman, 1971; Nemetz és Kusolitsch, 1999; Solt, 1971; Yule, 1950)

$$M = \frac{\sum_{i=1}^{n} \Delta y_i}{n},\ \sigma = \sqrt{\frac{\sum_{i=1}^{n} (\Delta y_i - M)^2}{n}}. \tag{4}$$

Azokat az eltéréseket tekintettük szignifikánsnak, amelyek az átlagtól legalább 2σ-val eltérnek. A 1/B-3/B ábrák mindegyikén tisztán kivehetőek a Δf görbének azok a pontjai, amelyek az $M \pm 2\sigma$ tartományon kívül esnek. Arra voltunk kíváncsiak, hogy milyen események következtek be a szövegben, amelyek ezeket a kiugrásokat eredményezték a görbéken.

Eredmények

Vizsgálatainkban különböző nyelveken írt irodalmi művek összehasonlítását végeztük. Ahhoz, hogy összehasonlítható eredményeket kapjunk olyan műveket kerestünk, amelyek több különböző nyelven is elérhetőek. Így esett a választás Kertész Imre SORSTALANSÁG című művére, amely angolul (FATELESS) is és németül is (ROMAN EINES SCHICKSALLOSEN) megjelent, valamint Rudyard Kipling THE JUNGLE BOOKS című műveire és ezek magyar fordítására (A DZSUNGEL KÖNYVE).

A választás azért esett ezekre a művekre és fordításaikra, mert szerkezetében lényegesen eltérő nyelvekről van szó. Aszerint csoportosítva, hogy a morfémákból hogyan képzi a nyelv a szavakat a három nyelv három különböző kategóriába sorolható. A német a flektáló, a magyar az agglutináló nyelvek csoportjába tartozik, míg az angol több különböző kategória eszközeit is felhasználja, így igazán egyikbe sem illik bele (O'Grady, 1993; É. Kiss, 1998; Kiefer, 1998; Kugler, 2000; Laczkó, 2000; Quirk et al., 1995; Uzonyi, 1996) (1. táblázat). A kérdés az volt, hogy a mondatok belső kohéziója, tehát a szintaktikai szabályok befolyásolják-e, s ha igen mennyiben az új szóalakok megjelenését, illetve származhatnak-e más forrásokból az eredeti és a mesterséges szöveg közötti eltérések.

Table 1. Kertész Imre SORSTALANSÁG, a mű angol és német nyelvű fordítása, Rudyard Kipling THE JUNGLE BOOKS és magyar fordítása. A második oszlopban a szöveg hosszát (szövegszó = 100 * blokk), harmadik oszlopban a különböző szóalakok, negyedik oszlopban pedig az egyszer előforduló szavak számát tüntettük fel az egyes művekben. A kapott értékeket összehasonlítva látható, hogy az angol szövegekben fordul elő a legkevesebb különböző szóalak és ezzel párhuzamosan a legkevesebb hapax legomena.

	blokk	**szóalak**	**hapax legomena**
Sorstalanság	561	14740	10253
Fateless	716	6710	3186
Roman eines Schicksallosen	719	9992	6043
The Jungle Books	1171	7452	3124
A dzsungel könyve	922	20362	13372

Az 1. táblázat értékei mutatják, hogy az egyes nyelvek sajátosságaiból, valamint a fordításból adódóan a szövegszók, a különböző szóalakok és az egyszer előforduló szavak száma között lényeges eltérések mutatkoznak az egymásnak megfelelő szövegek esetén. Ha azonban elemezzük a grafikonok kiugrásait (1-3. ábra), megkereshetjük az eredeti szövegben azokat a szakaszokat, amelyekben az újonnan megjelenő szavak lényegesen magasabbak, mint az a modell alapján várható lenne. A kérdés az volt, hogy mivel magyarázhatóak ezek a kiugrások, tehát a mi indokolja a különböző szóalakok szokatlanul magas számát és találunk-e olyan jellemzőjét a szövegnek, amelylyel leírhatóak ezek a hirtelen változások.

A kiugrások pontos helyének, a blokkok sorszámának meghatározását MS Excellel, azoknak az n*100 szövegszó hosszúságú szövegrészeknek a meghatározását, amelyekben ezek a kiugrások megjelentek pedig a szövegfeldolgozásra használt saját programmal végeztük. A szövegrészt ismerve vissza tudtuk azt keresni az eredeti

műben és magyarázatot tudtunk adni arra, hogy miért növekedett meg hirtelen az újonnan bevezetett szavak száma.

Table 2. Azoknak a blokkoknak a sorszáma, amelyekben az újonnan bevezetett szóalakok száma magasabb, mint az a modell alapján várható volt.

	Sorstalanság	Roman eines Schicksallosen	Fateless
Vili bácsi			43
csepeli üzem			71
indulás a vonattal		209	156
Auschwitzba érkezés	170		215
Buchenwaldba érkezés	262	337	329
reggeli készülődés, üzem	310	398	392
testének leírása			447
lelkiállapotának leírása		518	
kórház	429	552	
Pjetyka főz	459	587	
napi menetrend			618
haza indulás	510	651	

A SORSTALANSÁGban hat szignifikánsnak tekinthető eltérést találtunk és néztünk meg részletesen. Ezek a kiugrások valamennyien olyan esetben jelentek meg, amikor a szöveghez nem szervesen kapcsolódó, a korábbi eseményektől függetleníthető leírás jelent meg a szövegben. Ez a hat esemény a megjelenés sorrendjében a következő volt: megérkezés a koncentrációs táborba, megérkezés a második táborba, reggeli események és az üzem leírása, kórház leírása, Pjetyka főz, haza indulás (1. ábra, 2. táblázat).

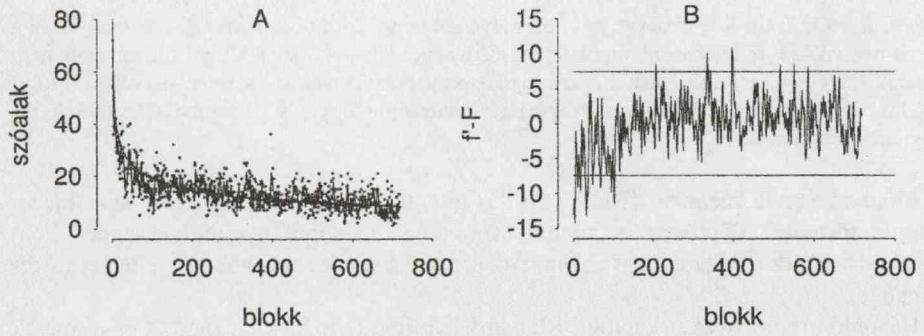


Fig. 3. Kertész Imre SORSTALANSÁG című művének német fordítása: ROMAN EINES SCHICKSALLOSEN. A német és a magyar nyelvű szövegben apró eltérésektől eltekintve a szövegnek ugyanazon a pontján emelkedett meg az újonnan bevezetett szóalakok száma.

A német nyelvű szövegben hét kiugrás található (2. táblázat, 2. ábra), amelyek közül az első nem a táborba érkezést, hanem egy korábbi eseményt, a vonatra szállást írja le. Várhatóan azért nem kaptunk a német szövegben újabb kiugrást a táborba érkezéskor, mert a vonatra szállás, a vonat leírására használt szavak nagyban fedik a

tábor jellemzésére használt szavakat. A második és a harmadik kiugrás ugyanannál a szövegrésznél következett be, mint a magyar szövegben. A német szövegben akkor jelenik meg a negyedik kiugrás, amikor egy leírás következik a főszereplő pillanatnyi lelkiállapotáról. Ez a leírás a magyar szövegben nem eredményezett szignifikáns eltérést. Végül az utolsó három kiugrás újra teljes egészében megegyezik a magyar szöveg kiugrásaival. (A német szöveg utolsó kiugrása még éppen az elfogadhatósági intervallumon belül esik, de ez várhatóan annak tudható be, hogy a digitalizálás során egy ének a magyar szövegben szótagolva került be, míg a német szövegben egybe írva.)

Az angol szöveg elemzésekor is hasonló eredményeket kaptunk (2. táblázat, 3. ábra). Olyan helyeken jelentkeztek a görbén kiugrások, ahol a műbe egy hosszabb lélegzetű leírás került. Ezek nagy része most is megegyezett a magyar (német) szöveg kiugrásaival, annyiban történt változás, hogy az angol szövegben összesen nyolc csúcs tekinthető lényeges eltérésnek a szöveg megszokott menetéhez képest. A magyar és a német nyelvű szöveghez képest megjelent a szöveg elején két kiugrás, amely további részletes leírást ad. A középső négy kiugrás megegyezik a másik két szöveg kiugrásaival, míg a két utolsó olyan leírás, amely csak az angol szövegben okozott szignifikáns eltérést.

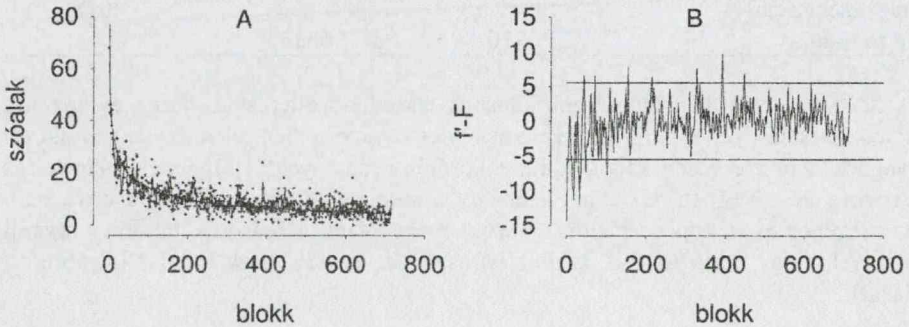


Fig. 4. Kertész Imre SORSTALANSÁG című művének angol fordítása: FATELESS. A magyar és a német nyelvű szöveghez hasonlóan olyan eseményeknél jelentek meg a kiugrások, amelyek nem képezik szerves részét a szövegnek, nem logikus következményei az előzményeknek, és a folytatáshoz sem kötődnek. Ezek a szövegrészek rendszerint egy-egy hosszabb lélegzetű leírás megjelenését jelentik.

Előzetes várakozásokkal (Balázs, 1985) ellentétben ezek a kiugrások nem fejezet határon történtek. Különös tekintettel arra, hogy az angol nyelvű szövegben nem ugyanott vannak a fejezet határok, mint az eredeti magyar szövegben és a német fordításban.

Hasonló eredményeket kaptunk Rudyard Kipling THE JUNGLE BOOKS és a művek magyar fordításának elemzésénél. Nem feltétlenül az újabb mese kezdetekor növekedett meg az újonnan bevezetett szóalakok száma, hanem sokkal inkább akkor, amikor egy hosszabb lélegzetű leírás jelent meg a műben. Ennek megfelelően egyes nem a dzsungelben játszódó történetben (The White Seal, Rikki-Tikki-Tavi, Toomai of the Elephants, The Miracle of Purun Bhagat, Quiquern), mivel színhelyük és témájuk rendkívül változatos a kiugrások egy-egy részletes leírás eredményei. A dzsungelről szóló történetekben is találtunk két lényeges kiugrást, de egyiket sem az adott mese

kezdeténél, hanem egyszer a királyi palota, míg a másik alkalommal a kincstár leírása okozta a szóalakok számának hirtelen emelkedését.

Az említett kiugrások tehát akkor következnek be, amikor a soron következő mondatok sem az előzményekhez nem kötődnek, sem a későbbiekhez való szervest kapcsolódást nem készítik elő. Olyan szövegrészek, amelyekhez nem találni olyan témát, amelyhez a bennük foglaltak kapcsolódnának. Az 1-3. ábrákon jól látható kiugrásokon túl ugyanezt támasztja alá az egyszer előforduló szavak vizsgálata is (4. ábra). Ugyanazokon a helyeken növekedett meg az egyszer előforduló szavak száma, ahol az eredeti műben szintén magas volt az újonnan bevezetett szavak száma. Ez a megfigyelés is arra enged következtetni, hogy a görbéken található kiugrások a szöveghez szervesen nem kapcsolódó részeknél jelennek meg.

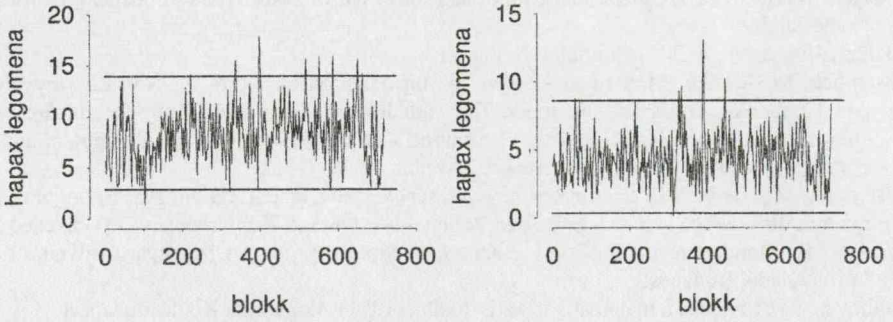


Fig. 5. Az egyszer előforduló szavak megjelenése ROMAN EINES SCHICKSALLOSEN (balra) FATELESS (jobbra) művekben. Az ábrán az átlag ±2*szórás jelző vonalakat a hapax legomena binomiális eloszlását feltételezve húztuk meg. A kiugrások azokon a helyeken jelentek meg, ahol az eredeti szövegben megnövekedett az újonnan bevezetett szóalakok száma.

Összegzés

Irodalmi művek statisztikai elemzésénél arra kerestük a választ, hogy mi okozhatja az eredeti mű és a szavak véletlenszerű megjelenését feltételező modellek közötti eltérést. A modelleknél azt tapasztaltuk, hogy eleinte felül becsülik az eredeti művet, míg vannak olyan helyek, amelyeken a modell által generált mesterséges szövegben kevesebb szóalak jelenik meg, mint az eredeti műben. Előzetes várakozásaink szerint ezek az eltérések egyrészt a szintaktikai szabályok következetes használatából adódhatnak, esetleg új fejezetek kezdetén növekedhet meg a szóalakok száma. Ezek a változások valóban megjelennek az újonnan bevezetett szóalakokat ábrázoló görbén, de csak kisebb, elsődleges kiugrásokat eredményeznek, ami nem nagyobb mint a zaj a görbén. Megválaszolatlan maradt azonban az a kérdés, hogy a másodlagos kiugrásokat mi okozza. Azt tapasztaltuk, hogy a másodlagos kiugrások szöveg szinten jelennek meg, akkor amikor a szöveg olyan leírásokat tartalmaz, amely sem az előzményekhez, sem a következő részekhez nem kapcsolódnak logikusan. Kétféle módszert is használtunk ennek igazolására. Első lépésként elemeztük a művek fordításait is, így rá tudtuk mutatni, hogy a mondaton belüli kohézió nem okozhatja ezeket a másodlagos kiugráso-

kat. Ezt követően megvizsgáltuk az egyszer előforduló szavak eloszlását, és az előzőhöz hasonló módon azt tapasztaltuk, hogy azokon a helyeken magas ezen szavaknak a száma, ahol a szöveghez alig kapcsolódó szövegrész jelenik meg.

Irodalom

Ashby, W. R. Bevezetés a kibernetikába (1972) Akadémiai Kiadó, Budapest

Baayen R. H.: Statistical Models for Word Frequency Distributions: A Linguistic Evaluation. Computers and the Humanities 26. (1993) 347-363.

Baayen R. H.: The Effect of Lexical Specialization on the Growth Curve of the Vocabulary. Computational Linguistics 22. (1996) 455-480.

Baayen, R. H. Word Frequency Distributions (2001) Kluwer Academic Publishers, Dordrecht, Netherlands

Balázs, J. A szöveg (1985) Gondolat, Budapest

Csernoch, M. Another Method to Analyze the Introduction of Word-Types in Literary Works and Textbooks, Conference Abstract, The 16th Joint International Conference of the Association for Literary and Linguistic Computing and the Association for Computers and the Humanities (2004) Göteborg University, Sweden

Csernoch, M; Hunyadi L. Szótípusok bevezetésének szabályszerűsége magyar és angol nyelvű nyomtatott szövegekben, Magyar Számítógépes Nyelvészeti Konferencia (2003) Szeged

É. Kiss, K. Mondattan. In É. Kiss, K., Kiefer, F. Siptár, P. (eds), Új magyar nyelvtan (1998) Osiris Kiadó, Budapest

Hajtman, B. Bevezetés a matematikai statisztikába (1971) Akadémiai Kiadó Budapest

Hoover D. L.: Another Perspective on Vocabulary Richness. Computers and the Humanities 37 (2003) 151-178.

Kiefer, F. Alaktan. In É. Kiss, K., Kiefer, F. Siptár, P. (eds), Új magyar nyelvtan (1998) Osiris Kiadó, Budapest

Kugler, N. Alaktan. In Balogh, J., Haader, L., Keszler, B., Kugler, N., Laczkó, K. Lengyel, K. (eds), Magyar grammatika (2000) Nemzeti Tankönyvkiadó, Budapest

Laczkó, K. Alaktan. In Balogh, J., Haader, L., Keszler, B., Kugler, N., Laczkó, K. Lengyel, K. (eds), Magyar grammatika (2000) Nemzeti Tankönyvkiadó, Budapest

Nemetz, T.; Kusolitsch, N. Guide to the empire of random (1999) TypoTEX, Budapest

O'Grady, W., Dobrovolsky, M. and Aronoff, M. Contemporary Linguistics, An Introduction (1993) New York: St. Martin's Press.

Quirk, R.; Greenbaum, S.; Leech, G.; Svartvik, J. A Comprehensive Grammar of the English Language (1995) Longman Group UK Limited, London and New York

Solt, Gy. Valószínűségszámítás (1971) Műszaki Könyvkiadó, Budapest, Hungary

Uzonyi, P. Rendszeres német nyelvtan (1996) AULA Kiadó Budapest, Hungary

Yule, G. U. An Introduction to the Theory of Statistics (1950) Charles Griffin & Company Limited, London, UK

Weöres Sándor költői nyelvének számítógépes feldolgozása

Nagy L. János[1], Alexin Zoltán[2]

[1] Szegedi Tudományegyetem, JGYTFK, Magyar nyelvi tanszék, Szeged, Boldogasszony sgt. 6., e-mail: nagy@jgytf.u-szeged.hu
[2] Szegedi Tudományegyetem, TTK, Informatikai Tanszékcsoport, Szeged Árpád tér 2., e-mail:alexin@inf.u-szeged.hu

Kivonat A cikkben bemutatásra kerül egy Weöres Sándor költői nyelvének számítógépes feldolgozásával foglalkozó projekt. A kutatás 1999 és 2002 között folyt, eredményeképpen kb. háromezer számítógépes oldalnyi korpusz jött létre, elemezve a HuMor magyar szóalaktani elemző programmal. Eredmény az adatbázis-formátum, amely lehetővé teszi a korpusz eltárolsát és a gépi feldolgozás megalapozását. Rendelkezésre áll Weöres szövegeihez a metrikai-ritmikai struktúra leírása. A kutatás távlataiban meghatározható a magyar nyelvű lírai textusok eltárolására alkalmas adatbázis-formátum; leírható a hangzás formalizálható adatainak digitális metakommunikációja; megadható a szükséges szoftverek száma és munkaerő szükséglete, – és megrajzolhatók a számítógépes elemzés korlátai.

Kulcsszavak: számítógépes szövegreprezentáció, természetesnyelv-feldolgozás, poétikai jellemzők adatbázisa

1. Bevezetés

Weöres Sándor költői munkásságának számítógépes feldolgozása 1999-ben kezdődött el a Szegedi Tudományegyetem Juhász Gyula Tanárképző Főiskolai Kar Magyar nyelvi tanszékén. [3]

Az előfeltételek között szerepel az az évtizedes filológiai kutatás, amely Weöres költészetének modern feldolgozását végezte, s 1991 és 1999 között sikeresen megvédett kandidátusi disszertációt, két monografikus elemzést (1996, 1998), számos tanulmányt és előadást eredményezett (Nagy L. János).

A számítástudományi előfeltételeket Alexin Zoltán kutatói műhelye, az Informatikai tanszékcsoport biztostotta, amely elismert magyar nyelvtechnológiai centrum. A tanszékcsoport kutatói a számítógépes szövegreprezentáció területén értek el eredményeket[1][2]. A nevükhöz fűződik a Szeged Korpusz elkészítése. A korpuszhoz használt szövegreprezentációs technológia teremti meg az alapot Weöres Sándor verseinek számítógépes feldolgozásához.

[3] A szerzők köszönetüket fejezik ki az Országos Tudományos Kutatási Alapnak: az ismertetésre kerülő kutatást az **OTKA T 029 431** sz. pályázttal támogatta.

A kutatást tanácsaival támogatta a nemzetközi hírű szövegkutató Petőfi S. János és a filológus akadémikus Fónagy Iván. Az alábbiakban a ismertetjük a kutatás fontosabb célkitűzéseit és eddigi eredményeit.

2. A projekt fontosabb célkitűzései

A munkaterv Weöres Sándor költői nyelvének számítógépes feldolgozását tűzte ki célul. Ehhez az alapcélhoz számos részcél kapcsolódott.

A *filológia* területén először azt kellett tisztázni, mit is jelent a "Weöres Sándor költői nyelve" terminus. A négyéves kutatási ciklus második felére, az MTA Irodalomtudományi Intézetében tartott előadás [4] diszkussziójában vált nyilvánvalóvá, hogy

- a virtuális kritikai kiadás létrehozása az első lépés, azaz az *Egybegyűjtött írások* és az *Egybegyűjtött műfordítások* szolgálnak az 'editio princeps' alapjául, s ezek korpuszához adódnak a Weöres halála után kiadott munkák, illetve a költő életében keletkezett, de az *Egybegyűjtött írások* és *Egybegyűjtött műfordítások* anyagába be nem került szövegek (részletesebben l. a [5] 3. fejezetében);
- a változatok (az *Egybegyűjtött írások* és *Egybegyűjtött műfordítások* előtt, illetve után publikált, kisebb-nagyobb eltérésekkel közölt szövegek) elemzése akkor valósulhat meg, ha ezt (az elbírálás folyamatában lévő) a célt is támogatja a kutatásfinanszírozás, - kétségtelen, hogy az igazi virtuális kiadás csakis eképpen valósítható meg;
- a változatok és az 'editio princeps' összevetése, a tanulságok adatokon alapuló, szolid összegzése feltétlenül szükségessé teszi a metrikai-ritmikai elemzések elvégzését, – kétségtelen, hogy ennek finanszírozása külön is szükséges (l. az előző bekezdést);
- várható bár, hogy a továbbiakban napvilágra kerülnek olyan szövegek, amelyek eddig kiadatlanok voltak, – jelentős számú ilyen textus előkerülése azonban nem valószínű.

Összefoglalóan elmondható, hogy a nálunk számítógépen (és CD-ROM-on) tárolt Weöres-szövegtár 'editio pronceps'-e a lehetőségek szerint teljes; míg a variánsokról ezt nem állíthatjuk.

A számítógépes feldolgozás alapvető munkafolyamatai következők voltak:

- mindenekelőtt szkenneléssel elérhetővé tettük a szöveganyagot a számítógép számára: a Recognita program használatával a kötetek betűtípusaihoz is alkalmazkodhattunk ("első változat");
- a szkennelt anyagot ellenőriztük és javítottuk, majd az eljárást megismételtük, – jellemző, hogy a metrikai-ritmikai elemzés is segített néhány rövidhosszú magánhangzó értékének meghatározásában a poétikai funkció(k)nak megfelelően ("második változat" és "harmadik változat");
- a javított és ellenőrzött textus lexikai anyagát a HuMor magyar nyelvi morfológiai elemző programmal elemeztettük, ezzel létrejött az elemzett változat;

- az elemzett anyagon gyakorisági vizsgálatot végeztünk nagyobb szövegegységenként, hogy összehasonlítható eredményekhez jussunk, azaz a lírai, a drámai és a prózai szövegeket külön-külön és együttesen is adatoltuk;
- az így nyert, gyakoriság szerint rendezett szóanyagon kvül megmaradt lexikai korpusz további finomítási feladatokat ad, a nyelvek szerinti tagolásban külön-külön tároltuk a képzelt, illetve az idegen nyelvű részhalmazt;
- a további célokat – eredeti célkitűzéseink szerint is – kb. hároméves munkával érhetjük el.

Jelenleg világosan látható, hogy a kutatási célt érdemes kissé kiterjeszteni. Eszerint az egyik bővítés az összehasonlítható metrikai-ritmikai adatbázis létrehozását, a másik a virtuális kritikai kiadás megteremtését eredményezheti. A két feladat össze is függ: az egyik a másik nélkül kevésbé jelentős. Egy-egy vers lexikai szövetének a változataiban magától értetődően vannak jelen a kicserélődő szavak, szórészek morfémáinak a hangzásban, a ritmusban bekövetkező variánsai. Nem is szólva azokról a (rövidebb) versekről, amelyeknek a létrejötte is közvetlenül összefügg dallamokkal, dallamtöredékekkel, – a kompozíciójuk pedig akár zenei műformákkal.

Ha van egyáltalán olyan költőegyéniség, akinek a lírájában a poétikai textus elengedhetetlenül, lényegében tartalmazza metrikai-ritmikai hangzását, azt a hangzást, amely csakis a jelentés csorbítása révén változtatható, – nos, Weöres Sándor éppen ez a poéta. Praktikus szempontból pedig: kár volna a megteremtett szövegbázistól kínált lehetőséget kihasználatlanul hagyni.

Ami pedig azt a kérdést illeti, hogy a Saussure utáni kor areferenciális gondolkodásmódja a véletlen, a szórtság, a játék „végtelen szöveg" fogalmával operál, s a retorikai megközelítésnek ad elsőbbséget a szemantikaival szemben a nyelvészeti kutatásban, – a Weöres Sándor-i „röpülő vers" éppen avantgárd gyökereivel, totalitásra törekvésével kínálkozik mintául. Mintául, amely számos különböző megközelítést visel el.

2.1. A tárgykörben kidolgozott elméletek, módszerek és eljárások

A tárgykörben alkalmazott nyelvészeti és poétikai elméletek, módszerek a korpusznyelvészetben ismertek: a *filológiai* feladatokban Gáldi Lászlóra név szerint is hivatkoztunk. [4] A metrikai-ritmikai elemzésben a szimultán verselésnek sajátos, Weöres Sándor által (Marsall Lászlónak küldött) levelekben megírt módszereit igyekeztünk követni. [5] A filológiai eredmények elérésében kulcsszerepük volt azoknak a hallgatóknak, akik speciálkollégium keretében ritmizálták a szöveganyagot; az adott metrikai-ritmikai struktúrák segítségével lehetőség nyílik a számítógépes elemzésben az igen távlatos, Weöres Sándor költészetén messze túlnyúló vizsgálatokra. Ezek a szövegfüggetlen struktúrák összevetéseiben ragadhatnak meg olyan belső összefüggéseket, amelyek a lexikai textus közbeavatkozása nélkül analizálhatják a formai egyezéseket és különbözőségeket.

[4] A Petőfi-szótár előszavában kövonalazott elvekről van szó, Gáldi László 1972. I: 7–18

[5] L. még Szuromi Lajos: *Szimultán verselés* Akadémiai Kiadó, 1990.

A *számítástechnikai* feldolgozás számára alapkövetelmény a biztonságra törekvés, az ellenőrizhetőség, az eljárások folyamatos tesztelése, az adatok pontossága. Ennek eredményeképpen meggyőző eredmény a gyakorisági elemzés, finomításán tovább dolgozunk (remélhetőleg az újabb OTKA-támogatás segítségével is). A további feldolgozás számára is nélkülözhetetlen a folyamatos kontroll: ebben az elemzésben is számos indexálási megoldás alapján lehetséges a számítógépes adatrendezés.

A költői nyelv feldolgozására magyar nyelvre kidolgozott adatbázis formátum nem áll rendelkezésre. A világban elérhető adatbázisok közül megvizsgáltuk azokat, amelyeket eredetileg angol nyelvű feldolgozásra fejlesztettek ki. A számunkra és a feladatnak leginkább megfelelőt, a TEI (Text Encoding Initiative)[6] konzorcium által ajánlott formátumot választottuk ki. Ennek a magyar nyelvű textus igényeihez adaptálása, a szükséges változtatások elvégzése áll a folytatás középpontjában.

3. A retorikus nyelvhasználat Weöres Sándor költészetében

Weöres Sándor tipikus XX. századi költő, azzal a retorikus sajátsággal, hogy hatásával kiszakítja olvasóját a mindennapok prózai világából: „emberfölötti áramok"-at alkalmaz. Ezek az emberfölötti áramok a vers hangzásvilágának hatásaként érvényesülnek, ahogyan maga Weöres fogalmaz: „a vers tartalmilag fogalmi, formailag auditív művészet". A költő törekvéseiben jelen van a totalitásra törekvés is: a teljes univerzumot kívánja megragadni, az emberiség történelmét, kultúráját a tájak és a korok egységében, ugyanakkor mindenfajta kötöttségtől függetlenül.

Nagy L. János harmadik Weöres-könyvében [7] [5] a retoricitás klasszikus és modern fogalmából indul ki. Legfontosabb elemzései a chiasztikus struktúrákat, a költői hangzásokat, a szövegvariációkat és a többrészes kompozíciókat kutatják. Az értelmezések a verselés metrikai-ritmikai szerkezetének, az értelmi és ritmikai hangsúlyoknak a jelentésképző szerepét kutatják.

Nagy L. János meghatározott elméleteket alkalmaz: a funkcionális stilisztikát, a szemiotikai textológiát, a szimultán verselést. Az értelmezés tényeihez és a következtetésekhez a szintaktikai, szemantikai és pragmatikai oszcillációk segítségével jut el. A formailag is megragadható azonosságok és különbségek szolid alapot adnak annak megközelítéséhez, ami a számítógépes elemzésre támaszkodva, de csakis a számítógép nélkül, a befogadói intellektus interpretáló tevékenységében ragadható meg.

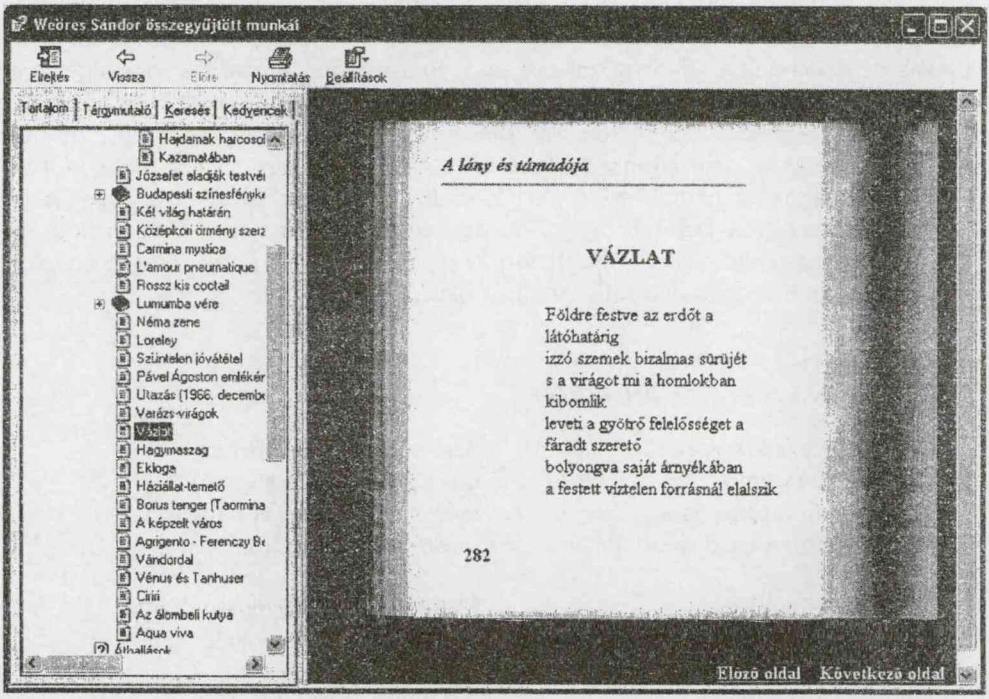
[6] A TEI konzorcium honlapja: `http://www.tei-c.org`

[7] Ezt a könyvet a kiadó és a szerző Weöres 90. születésnapjának tiszteletére szánta, 1913. június 22.

4. Eredmények és további lehetséges kutatási irányok

Az eddigi eredmények a XX. századi költői nyelv feldolgozásában korábban hasznosított eljárásokra épülő módszerekkel születtek, ilyen módon sokoldalúan összehasonlítható és felhasználható tanulságaik vannak. A további célok eléréshez annak függvényében juthatunk el, amilyen mértékben támogatja az OTKA (és más lehetséges finanszírozó) a terveket, pl. az informatikai és adatbeviteli tevékenységek honorálásával.

Az összegyűjtött korpuszhoz Papp Endre programtervező matematikus hallgató diplomamunkája keretében Alexin Zoltán irányításával egy webes megjelenítő programot készített. Ennek egy képernyője látható az 1. ábrán.



1. ábra. A Weöres korpusz webes megjelenítő modulja

Az eredmények alapján az OTKA pályzat kezdetén megfogalmazott célok jelenleg is érvényesek, azaz a folytatásban is követendők:

a) Weöres Sándor 'editio princeps'-ének gyakorisági szótára befejezéséhez szükség van kb. kétévi munkára és egyévi ellenőrzésre; a tevékenység finanszírozására újabb OTKA-pályázatot nyújtottunk be;
b) a szövegvariánsok digitalizálása megteremti a virtuális kritikai kiadás textusát, ezzel az 'editio princeps' kiegészül a szövegkörnyezettel;

c) a metrikai-ritmikai elemzés adatait számítógépbe visszük (ez a tevékenység csak hallgatók bevonásával képzelhető el); erre irányul a pályázat második főiránya;
d) a számítógépes feldolgozás fentebbi lehetőségei közül a kiválasztott adatbázist adaptáljuk a magyar nyelvű szövegek igényeihez; az adatokat bevisszük és a feldolgozásra előkészítjük: erre irányul a pályázat harmadik célcsoportja.

A 2002-ben lezárult OTKA vizsgálatban sajátos feldolgozási problematika jelent meg. Weöres Sándor ugyanis abban is kitűnt kortársai közül, hogy gyakran használt az írásaiban az élő (és holt) nyelvek szókincsén kívül képzelt nyelvű lexikai anyagot. Ilyen módon a lexikai-morfológiai elemzés a feldolgozáskor nyilvántartotta ugyan ezeket az adatokat, de megelégedett a regisztrálásukkal, nem elemezhette a szóalakok felépítését.

Ennek a lexikának van "eredetiben és fordításban" egyaránt közölt része, pl. az 1944-es *Barbár dal* (alcíme: *Képzelt eredeti és képzelt fordítás*). Az 1960-as évek végén ennek a két változatnak az egymásra vonatkoztatása segítségével a jelenleg Párizsban élő *Nyéki Lajos* elkészítette a "képzelt eredeti" szótárát is. Ugyancsak egymás mellett közölte Weöres az 1946-os *Arany kés forog* (korábbi címe: *Örvényben, naur glainre iki kezdetű*) kétféle textusát, ennek azonban szótárszerű feldolgozása nem készült. A két említett vers publikálása abban tér el egymástól, hogy a laptükörben a *Barbár dal* nyomtatott formában a képzelt nyelvű anyag mellé teszi a képzelt fordítást, az *Arany kés forog* pedig a magyar textus mellé helyezi a képzelt nyelvű sorokat.

BARBÁR DAL
(Képzelt eredeti és képzelt fordítás)

Dzsá gulbe rár kicsere
áj ni musztasz emo
áj ni mankütvantasz emo
adde ni maruva bato! jaman!

Szél völgye farkas fészke
mért nem őriztél engem
mért nem segítettél engem
most nem nyomna kő! ajaj!

Ole dzsuro nanni he
ole csilambo ábábi he
ole buglo iningi he
lünlel dáji he! jaman!

Könnyemmel mosdattalak
hajammal törölgettelek
véremmel itattalak
mindig szerettelek! ajaj!

Vá pudd shukomo ikede
vá jimla gulmo buglavi ele
vá leli gulmo ni dede
vá odda dzsárumo he! jaman!
(1944)

Földed tüskét teremjen
tehened véres tejet adjon
asszonyod fiat ne adjon
édesapád eltemessen! ajaj!

Képzelt ősi kultúra képzelt ősi nevei jelennek meg a *Mahruh veszése* című szövegben (1952): bevezetést, 101 négysorost és a *Negyven király éneke* című, hat tömbre tagolódó befejező verset tartalmaz. Weöres a képzelt lantost *Rou Erou-*

nak, azaz *Bíbor Lángnak* nevezi. A négysorosok névanyagából néhány példa: *Kartiabh-todarh* (1.), *Lulobh* (4.), *Zölülügh* (11.), *Jöhpruk* (50.), Nuantompank (94.) stb. A tájnevek és személynevek egyszer-kétszer fordulnak elő mindössze.

Avantgárd szövegeiben Weöres gyakran tördeli morfémákra, morfémarészekre a szavakat; a hangzási kísérletekben számos nonszensz textust is ír: *Tapéta és árnyék* (1963), *Egérrágta mese* (1960), *Az áramlás szobra* (1944), *Az égő szótár* (1967), *A megmozdult szótár* (1949-53), *A levegő morzsái* (1978). (L. Károlyi Amy: Weöres Sándor és az avantgárd. *Holmi*, 1990: 1003.) A *Kilencedik szimfónia* (1955-60) *Finale* tételében az 1956-os forradalmat leverő orosz páncélosokat idézi: *"füstöl a tábornok / csikorgó szeke- / recsegő kere- / köhögő mene- / kürtöl a kolosszus / a kolosszusnak a"*. A *Táncdal* (1942) az egyik legismertebb hangzó nonszensz (*panyigai panyigai panyigai / ü panyigai ü*).

A *Wawiri* (1942) című gyűjtemény darabjaihoz hasonlít Weöres *Hangcsoportok* című háromrészes kompozíciója (1941), részei: *Puha, forró hangok (Ange amban ulanojje...)*; *Gyors, gyöngyöző, vidám hangok (Vikulili hejriri sziggaga...)* és *Áradó, sugárzó hangok (Khúnái áfháiszthái mengoh...)*. A francia *écriture automatique* ('automatikus írás') módszerével is jó néhány Weöres-szöveg született, pl. In memoriam Devecseri Gábor (jellemző az alcím: *Három repedezett szikla- vagy felhőgomoly-alakzat)*. A második rész (*Post Mortem*, 1971) egyik sora: *"ílyenkor elborít geometriai a az"*.

Összefoglalóan megállapítható, hogy a poétikai elemzésben pontosabb képet nyerhetünk majd ezeknek a lexikai elemeknek a szerepéről is. A hangzó lexikai elemekre is érvényes Weöres egyik nyilatkozata: "Mindig ugyanúgy viadalban, viaskodásban voltam vele, mint mindennel, ami érdekelt ...hogy milyenek a hangzói, milyen a struktúrája, ez érdekelt, ez foglalkoztatott." (A költészet hivatása. Tardos Júlia rádiós beszélgetése Weöres Sándorral. 1972. június 25. In: Domokos Mátyás (szerk.): *Egyedül mindenkivel*. Szépirodalmi, 1993: 216.)

4.1. A távlatok a következő *filológiai* eredményekkel kecsegtetnek

- Létrehozható az első olyan magyar nyelvű verskorpusz, amely a lexikával együtt tárolja a metrikai-ritmikai, hangzási annotációkat.
- Ez lehetőséget nyújt a lexikai és a metrikai-ritmikai oldal együttes megjelenítésére egy alkalmas szoftverrel.
- Mindez könnyen kezelhetővé, kereszthivatkozásokkal is nyitottá teszi az eddig mintegy 3000 számítógépes oldalnyi anyagot.
- A változatok és az 'editio princeps' lexikai és ritmikai tényezői közelebb hozzák a befogadóhoz és a kutatóhoz a költői alkotófolyamat fázisait.
- A kutatás egyértelműen és pontosan meghatározhatja a szövegváltozatok közötti különbségeket.
- Meghatározható, melyek azok a poétikai jegyek, amelyeket érdemes Weöres költészetében figyelembe venni; – ugyanakkor melyek azok, amelyeket más költők esetében is figyelembe kell venni; - és melyek azok, amelyek vizsgálata nem érdemel figyelmet.
- Meghatározható, hogy pontosan mely poétikai jegyek (és értelmezésük) kutatható számítógéppel, melyek pedig nem: azaz meghúzhatók a gépi elemzés kompetenciájának határai.

A korpusznyelvészet jelentős informatikai lehetőségeiből:

- Megteremthető az első olyan magyar verskorpusz, amely szabványos adatbázis formátumban tárolja egy nemzetközi hírű magyar költő összes textusát.
- Nagy valószínűséggel megtervezhetővé teszi magyar verskorpuszok számára az optimális adatbázis-formátumot: adatbázis-változatokat segítene elő.
- Szoftverek készülhetnek, amelyek támogatják a rögzítési, feldolgozási munkát.
- A korpusz létrehozása és a szükséges szoftvermennyiség és -minőség tapasztalatokat kínálna az adott célú korpuszok előállításának optimalizálására.
- Országosan terjeszthető CD-ROM változatban nemzeti kulturális értéket teremtene (beleértve angol, német stb. nyelvű ismertetését).

Hivatkozások

1. Alexin Z., Csirik, J., Gyimóthy, T., Bibok K., Hatvani, Cs., Prószéky, G., Tihanyi, L.: *Manually Annotated Hungarian Corpus*, in Proceedings of the Research Note Sessions of the 10th Conference of the European Chapter of the Association for Computational Linguistics EACL'03, Budapest, Hungary, 53–56 (2003).
2. Csendes, D., Hatvani, Cs., Alexin, Z., Csirik, J., Gyimóthy, T., Prószéky, G., Váradi, T.: *Kézzel annotált magyar nyelvi korpusz: a Szeged Korpusz*, Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2003), Szeged, Magyarország, 238–245, (2003).
3. Nagy L. János: *Költői szótár a XXI. században*, előadás a Magyar Alkalmazott Nyelvészeti Konferencián, Pécs, Magyarország, 2001. április 5. p. 8.
4. Nagy L. János: *Filológiai problémák Weöres Sándor költői nyelvében*, előadás az MTA Irodalomtudományi Intézetben, Budapest, Magyarország, 2001. április 11. p. 24.
5. Nagy L. János: *A retorikus nyelvhasználat Weöres Sándor költészetében*, Akadémiai Kiadó, Budapest, 2003. 284 oldal + bibliográfia
6. Nagy L. János: *Poésie musicale ou musique poétique? Sur le compositions*, S. Weöres, F. Sebő: *Revue d'Études Francaises*, Budapest, **3** 181–195, (1998)
7. Nagy L. János: *A szintaxis poétikájához*, In Schaeffner A., Bene K., Madácsy P. (szerk.): *Új tendenciák a komparatisztikában IV., Nouvelles tendences en littérature comparée IV.* JGYTFK, Szeged-Amiens, 291–304, (2004)
8. Nagy L. János: *Chiasme: a semiotic outline, Sprachteorie und germanische Linguistik.*, Kossuth Egyetemi Kiadó, Debrecen, **1** 19–35, (2004)

Az iskolai idő értékelése nyolcadik osztályosok érvelő fogalmazásainak tartalomelemzése alapján

Huszár Zsuzsanna[1], Dr. Sramó András[2]

[1]PTE BTK Tanárképző Intézet, 7624 Pécs, Ifjúság útja. 6.
huszped@tki.pte.hu,
[2]PTE KTK Gazdasági Informatika Tanszék, 7622 Pécs, Rákóczi út 80.
sramo@ktk.pte.hu

Kivonat: Jelen írásunk az előző évi konferenciára készített előadásunk (*Idői struktúrák feltárása kvalitatív és kvantitatív szövegelemzéssel*) szerves folytatása; végigviszi és összegzi a 2003-as prezentációban példával illusztrált tartalomelemzést a nyolcadik osztályosok érvelő fogalmazásainak részmintájára. Az idői struktúrák vizsgálatának korábban vázolt szempontrendszerén túl az új előadás bemutatja a szövegjellemzők bináris kódolását, mint a számítógépes feldolgozás manuálisan előállított bemenetét. Az előadás ezen bemenet alapján lehetséges gépi ábrázolásmódokat mutat be, és rávilágít ezen újszerű ábrázolásmódok kutatásban betöltött funkciójára. A kivonat összegzi a cikk legfontosabb mondanivalóját.

Kutatásunkban az időt Durkheim, Gell és mások nyomán kulturális konstrukcióként tételezzük [1]. Társadalomtudományi közelítésben sem hagyható azonban teljességgel figyelmen kívül az idő fizikai természete, hiszen a társadalmi és iskolai élet szerveződésében jelentős szerepe van. Felidézzük, hogy az idő természettudományos fogalmának fizikai alapját három ciklikus mozgás adja: a Föld keringése a Nap körül (év), a Hold keringése a Föld körül (hónap), s a Föld keringése a saját tengelye körül (nap). Ezt a fizikai meghatározottságot, mint az idői léptékre való nyelvi utalást vesszük csak tekintetbe. Az idő rétegzettségére utalva figyelembe vesszük Evans-Pritchard „ökológiai idő" és „strukturális idő" megkülönböztetését [2], elfogadva a strukturális idő korlátozott-szimbolikus keretekhez kötött érvényességét. Az ebben az értelemben vett „társadalmi idő" tartalmára vonatkozó érvelések között méltányolva Leach álláspontját, aki szerint a társadalmi időfelfogást az az általános emberi tapasztalat tölti meg tartalommal, amely szerint az emberi élet visszafordíthatatlan változások lineáris tapasztalata, egyszersmind ismétlődő és ciklikusan visszatérő folyamatok tapasztalata is egyben. Leach felfogásában időkategóriánk ezt a kettős tapasztalatot olvasztja egybe [3].

Az antropológiai megközelítés korlátairól szólván Iténau álláspontjára utalunk, aki leszögezi, hogy az antropológia nem rendelkezik olyan eszközökkel, amelyekkel képes lenne elvonatkoztatni az időt annak társadalmi kifejezési formáitól. Iténau lehetséges megközelítési módként a kontextus vizsgálatát emeli ki [4].

Az időhöz való társadalmi viszonynak a nyelv az egyik kitüntetett hordozója, maga a nyelvhasználat pedig a társadalmi kontextus egyik lényeges eleme. Kiindulópontunk, hogy a társadalmi időszemlélet és időkezelés nyelvi bázison megragadható, hogy az iskolai fogalmazások utalnak az idővel való társadalmi bánásmódra, és hogy az iskolai

fogalmazások elemzésével a társadalmi időkezelésről és időfogalomról új ismeretekhez juthatunk.

Kutatásunk szövegbázisát a 10-16 évesek iskolai fogalmazásainak a Szegedi Tudomány-egyetem létrehozott reprezentatív mintája képezi. Jelen előadásunk nem a corpus egészén, hanem annak egy részmintáján kvalitatív metodika alkalmazásával három fő szempontot – idői, értékelési és téri szempontot – érvényesít, és ezek kapcsolódását prezentálja:

- a szövegek "időarányait" mint a múltra, jelenre, jövőre vonatkozó tartalmak előfordulási gyakoriságát tekinti;
- a szövegekben foglalt "időértékelést" mint a múlthoz, jelenhez, jövőhöz kapcsolódó tartalmak pozitív, negatív semleges vagy ambivalens érzelmi megítélését értelmezi;
- s megvizsgálja a pozitív, negatív, semleges és ambivalens értékelésekhez tartozó topológiai kategóriák előfordulását valamint a múltra, jelenre, jövőre vonatkozó közlések térbeli vonatkozásait.

Iskolai fogalmazások tartalomelemzésével vizsgáljuk, hogy miként írható le „az iskola saját ideje", és miként tükröződik az iskolai idő értékelése a nyolcadikosok fogalmazásaiban. Az iskola saját idejére Meleg Csilla nyomán az időszociológia és a szervezetszociológiai egymásra vonatkoztatásával tekintünk [5].

Linearitás és alienaritás szövegszerkezeti szempontból is értelmezhető [6], és a mondat időszerkezete vizsgálható [7].

Elemzésünkben a nyelvészeti és neveléstudományi megközelítés hátterében az időszociológia [8], az időantropológia és a szervezetszociológia szempontjait érvényesítjük Elemzésünk egységei a mondatok, illetve tagmondatok, objektumai az egyes fogalmazások. Az aspektualitás vizsgálatára [9] korlátozott módon térünk ki. A tartalomelemzés alapján tulajdonságokat rendelünk az egyes fogalmazásokhoz, és Galois-gráfok előállításával ábrázoljuk [10, 11] az egyes fogalmazások, illetve a fogalmazások adott csoportjainak jellemzőit. A szövegjellemzők bináris kódolása egyelőre a számítógépes ábrázolás bemeneteit jelenti.. Vizualizációs technikák alkalmazása elmélyíti, kiegészíti a kvalitatív elemzést, s a szocio-demográfiai háttér néhány tényezőjének figyelembevételével finomítja azt az általános képet, amely szerint az iskolai idő megítélése annál pozitívabb, minél távolabbi helyekhez kapcsolódik az iskolai élmény. Miközben alátámasztják eredményeink a tanórán, és iskolán kívüli pedagógiai tevékenységek fontosságát és hatását, az idői és szervezeti viszonyok egymásra vetítésével utalnak a társadalmi idő pedagógiai szempontból figyelembe veendő újabb rétegeire is.

A következő lépésben megkerülhetetlen kutatási feladatunknak tekintjük a számítógépes tartalomelemzést, a LAS Verticum időmodulja [12] használati tapasztalatainak beépítését.

Bibliográfia

1. Gell, Alfred: Idő és szociálantroplógia. In: Fejős Zoltán (szerk.): Az idő antropológiája. Osiris Kiadó, Budapest, (2000) 13-34.
2. Evans-Pritchard, E. E.: The Nuer, a description of the modes of livelihood and political institutions of a Nilotic people, Oxford, At the Clarendon Press, (1940)

3. Leach, E.R.: Rethinking Anthropology. London, Athlone Press (1961)
4. Iténu, André: Szinkronizáció az orokaiváknál. In:Fejös Zoltán (szerk.): Az idő antropológiája. Osiris Kiadó, Budapest, , (2000) 247-268.
5. Meleg Csilla: Az iskolai idő. Elhangzott október 14-én Budapesten az FKI-ban rendezett felsőoktatási kutatási konferencián (2004)
6. P. Balázs Géza: A linearitás és az alinearitás mint szövegszerkezeti alaptörvényszerűség. In: Horváth Katalin - Ladányi Mária (szerk.): Elemszerkezet és linearitás. A jelentés és szerkezet összefüggése. ELTE BTK, Általános és Alkalmazott Nyelvészeti Tanszék, Budapest, (1998) 7-13.
7. Kiefer Ferenc: A mondat időszerkezete. In: Jelentéselmélet. MTA Nyelvtudományi Intézet. Corvina, (2000) 248-274.
8. Gelléríné Lázár Márta (szerk.): Időben élni. Történeti szociológiai tanulmányok. Akadémiai Kiadó, Budapest (1990)
9. Wacha Balázs: Időbeliség és aspektualitás a magyarban. Nyelvtudományi Értekezések 149.sz. Akadémiai Kiadó, Budapest, (2001) 101 old.
10. Takács Viola: Galois-gráfok pedagógiai alkalmazása. Iskolakultúra- könyvek 6. Iskolakultúra, Pécs, (2000) 197 old.
11. Szigeti Márton: Galois-gráfok rajzolása számítógéppel. In: Takács Viola: Baranya megyei tanulók tudásstruktúrái. Iskolakultúra könyvek 20.Iskolakultúra, Pécs, (2003) 169-190.o.
12. Echmann Bea: A LAS Verticum narratív pszichológiai tartalomelemző rendszer időmoduja. In: Alexin Zoltán - Csendes Dóra (szerk): Magyar Számítógépes Nyelvészeti Konferencia 2003, SZTE Informatikai Tanszékcsoport, Szeged, (2003) 290.

Többnyelvű közmondás-adatbázis

Hrisztova-Gotthardt Hrisztalina[1]

Pécsi Tudományegyetem
Bölcsészettudományi Kar, Nyelvtudományi Doktori Iskola
Postai cím: 7632 Pécs, Olga u.1.X.32
E-mail: xpucu@freeweb.hu

Kivonat A cikk egy többnyelvű közmondás-adatbázist mutat be. A hálószerű séma szerint felépülő WEB-alapú adatbázis részletesen prezentálja a benne szereplő közmondások jegyeit és a hozzájuk tartozó fontos információkat, valamint lehetővé teszi a különböző nyelvekben levő közmondások közötti szemantikai kapcsolatok létrehozását. Az adatbázis általános hozzáférhetősége, részletes keresés, kiegészítés és javítás lehetősége segítheti a nyelvész- és folklór-kutatók, a tankönyvkészítők, a fordítók és a nyelvtanárok munkáját, akik tudományos, szerkesztői, fordítási és egyéb munkájuk során korpuszként, illetve gyors tájékozódáshoz és ellenőrzéshez használhatják.

1 Bevezetés

A disszertációs projektem keretében, amely bolgár, magyar és német közmondások kontrasztív elemzését a világ nyelvi képe fényében tűzi ki célul, felmerült a kutatásom során használható adatbázis hiányának problémája. Munkám során olyan hipotéziseket kívánok bizonyítani, amelyekhez egy ilyen adatbázis megléte elengedhetetlen. Jelenleg azonban még nem létezik számítógépes közmondás-adatbázis, amely a többnyelvűséget biztosítja. A kétnyelvű szótárak is – kétnyelvű adatforrásként – e téren hiányosnak bizonyulnak; a bennük szereplő parömiák száma igen csekély, a megfelelések nem ritkán pontatlanak vagy számuk nem teljes. A többnyelvű közmondás-gyűjtemények gyakran csak néhány – bizonyos szempontok alapján kiválasztott – nyelvet vesznek alapul, és ilyen módon leszűkítik azt a lehetőséget, hogy különböző nyelvi csoportokba és családokba tartozó nyelveket hasonlítsunk össze. Nikolaj Ikonomov (bolgár, albán, görög, román, orosz, szerb és török közmondások) és Szergej Vlahov (bolgár, angol, francia, német és latin közmondások) gyűjteményei többnyelvűségre törekednek, mégis egy központi nyelv – a bolgár – köré csoportosulnak, amelyet vagy csak a szomszédos, vagy csak a „világnyelvekkel" ütköztetnek. A fentiek értelmében e projekt során egy többnyelvű közmondás-adatbázis kidolgozása a célom, melyet a következőkben ismertetek.

2 Adatbázis: célok és követelmények

Egy leendő számítógépes közmondás-adatbázis több célt is szolgálhat. Korpuszként elsősorban kutatási célokra használható. Ahogyan Schneider is említi, egy jól strukturált és kidolgozott korpusz nagy segítséget nyújthat, és hasznos alapot biztosíthat hipotézisek bizonyítása, a nyelv részletes, empirikus-tudományos leírása során (Schneider 2000)[6]. Másodsorban egy többnyelvű adatbázis fontos szerepet játszhat az anyanyelv és idegen nyelv oktatásában – tankönyv és tanítási anyagok készítői információforrásként használhatják, amely az adott nyelv legelterjedtebb közmondásait és más nyelvű megfeleléseit tartalmazza. Nem utolsó sorban hasznát vehetik a különböző szótárak szerzői, attól függetlenül, hogy egy-, két-, vagy többnyelvű szótárról van-e szó. Ebben az értelemben a fordítók is kihasználhatják az előnyeit, mivel a célnyelvben nem mindig áll rendelkezésükre „szó-szerinti", teljes azonosságot mutató parömia.

Az adatbázis-követelményeket illetően, a következőket tűztem ki legfontosabbnak és elengedhetetlennek:

- Az adatbázis struktúrája legyen átlátható
- Sokoldalúan legyen lekérdezhető
- Legyen aktualizálható: adjon lehetőséget törlésre, kiegészítésre, javításra
- Az általános hozzáférhetőség miatt legyen WEB-alapú

Dobrovol'skij szerint egy jó adatbázist nem csak a reprezentatív adatok elegendő mennyisége tesz ki, hanem az egyes adatok részletes leírása (Dobrovol'skij 2002:430)[3]. Emiatt a fenti négy pont kiegészül az alábbi ötödikkel:

- Az adatbázis az adatok (a közmondások) lehetőség szerinti legszélesebb körű leírását tartalmazza

3 Az adatok

Mivel a disszertációs projektem keretében a három – bolgár, magyar és német – nyelvben szereplő lehetőleg legismertebb és legelterjedtebb közmondásokat szándékozom elemezni és összevetni, ennek létrehozásának első szakaszában az úgynevezett „parömiológiai minimumokkal" fogok dolgozni. A három elemzendő nyelv közül csak a magyar rendelkezik kidolgozott parömiológiai minimummal, amely 158, az adatközlők legalább 90%-a által ismert közmondást foglal magában (lsd. Tóthne Litovkina, Anna 1996)[7]. A magyar közmondás-minimumot alapul véve, felépül a leendő adatbázis, és később kiegészül német és bolgár nyelvű adatokkal. A közmondások várható száma a kevésbé ismertekkel együtt több ezerre tehető nyelvenként.

4 Adatstruktúra

4.1 Szemantikai rész

Az általam folytatott elemzés fő tárgya a három nyelv közmondásaiban kibontakozó világ nyelvi képének feltárása, illetve rekonstruálása. Ezen belül különös figyelmet fordítok a teljes azonosságra, az ekvivalens formák hiányára, és a gondolatbeli megfelelés, de ugyanúgy a felszíni (képbeli) eltérés bemutatására. A kognitív nyelvészet téziseire hivatkozva a közmondásokat a világról szóló ítéletekként definiálhatjuk (Bańczerowski a,b)[1][2]. Így az analízis során három alapvető kérdésre keresünk választ:

- Miről léteznek ítéletek az adott nyelvben?
- Mi egy-egy ítélet mondanivalója?
- Hogyan (milyen nyelvi kép segítségével) juttatja kifejezésre a három nyelvközösség az ítéletet?

A kérdéseket csak fordított sorrendben tudjuk megválaszolni – a felszíni struktúrától kiindulva (a nyelvi megvalósítás) a mély struktúra (kognitív szint) felé haladva. A legtöbb különbség a felszínen várható, mivel a három nyelvközösség a történeti, társadalmi, kulturális és nyelvi fejlődésénél és rendszerénél fogva más-más képet vesz igénybe az adott ítélet megfogalmazására. E hipotézis bizonyítására mindhárom nyelvből a parömiológiai minimumba tartozó közmondásokat szemantikai kategóriák útján visszavezetjük kognitív eredetükhöz, amely magában foglalja a fizikális tapasztalatok és mentális élmények által a világról megszerzett tudást és ennek a tudásnak a nyelvben történő kifejeződését. A visszavezetés struktúrája a következőképpen ábrázolható: (ábra 1).

Az adatbázis struktúra reprezentációs módja matematikailag egy irányított, véges, egyszerű gráf. A gráf ábráján a legalsó szinten találhatók a konkrét közmondások, ezek közvetlen elődei pedig az adott közmondás egyes jelentései. A gráfról leolvasható a szemantikailag rokon közmondások relációja, azaz ha két közmondás közötti útvonal „elég rövid", akkor azok jelentései között szoros rokonság áll fenn.

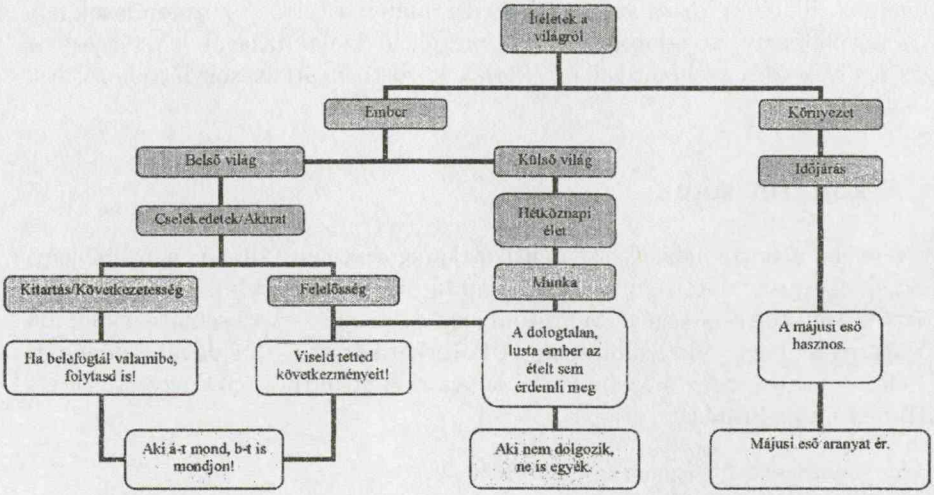
A fentiekben bemutatott módszer szükségessé teszi az ítéletek fogalmi kategóriákra való bontását, a közmondások kategóriákba való besorolását, és az adatbázis szemantikai részének létrehozását. Az így létrehozott „szemantikai" közmondás-adatbázis szolgál alapul a további elemzésekhez.

4.2 Közmondások eredetéről szóló információk

Írásbeli forrásokban fellelhető eredet is elengedhetetlen egy közmondás pontos leírásánál. Latin, középkorból származó, bibliai stb. eredetű parömiáknál fontos ennek az információnak a megadása.

4.3 Forrás

Közmondás-gyűjteményekből, különböző írásbeli vagy szóbeli szövegekből vett parömiáknál leírandó(k) a pontos forrás(ok).



1. ábra. A közmondások szemantikai rekonstrukciója (Az XML adatbázis egyszerűsített vizualizált kivonata)

4.4 Minősítések (stílus, beszédszándék, kor, tájnyelv)

Az adatok részletes leírásánál nélkülözhetetlen ezen jegyeknek a jelölése, például: *irod.* stílusértékként, *táj.* mint annak a jelölése, hogy egyes közmondások csak bizonyos régiókban ismertek, *rég.* mint annak a jele, hogy azok csak régi forrásokban találhatók stb.

4.5 Formai jegyek

Nem ritka jelenség az, hogy egy közmondásnak több formai változata/variánsa is létezik. A variánsokat a szótárak és a gyűjtemények szögletes zárójellel jelölik: *Embert szaván, ökröt szarván. [Ökröt szarvánál, embert szavánál.]* (O. NAGY 1999:504)[5] Hasonlóképpen sok közmondásnak létezik hosszabb és rövidebb alakja is. Az a rész, amely nem tartozik szükségszerűen a közmondáshoz, a gyűjteményekben kerek zárójelek között szerepel: *(A) szegény embert (még) az ág is húzza.* (O. NAGY 1999:115)[5] Számos közmondásnál mind a két lehetőséggel számolni kell. Ezek is abszolút módon szükségesek az adatok teljes leírásához.

5 Nyelvek közti megfeleltetések

Az eddigiekben egy-egy nyelv közmondásainak reprezentációját ismertettem, azonban olyan adatbázis-struktúrára van szükség, amely több nyelvből származó megfelelő parömiákat kapcsol össze. Az egyes adatbázis-lekérdezések

eredményeképpen láthatóvá kell válnia a teljesen vagy részben azonos formáknak, illetve a közös szemantikai tartományba tartozó közmondásoknak. Ez a követelmény az adatbázis séma megfelelő kialakításával lehetséges: az egyes nyelvek és a szemantikai kategóriák közötti hivatkozások létrehozásával teljesíthető.

6 A konkrét séma

Nyelvenként a közmondások szemantikai kategóriái egy fájlban, a többi jegy, érték, ill. az egyéb adatok egy másik fájlban tárolódnak. A két fájl közötti kapcsolatokat azonosítók segítségével valósítom meg. Erre a hálószerű sémára leginkább az XML nyelv bizonyul alkalmasnak. A következő ábra egy példával szemlélteti az ítélet-domének, azaz a szemantikai kategóriák struktúrájának egyszerűsített XML-beli megvalósítását.

```
<? xml version="1.0" encoding="ISO-8859-2" ?>
<ítélet>Ítéletek a világról
 <ítélet>Ember
  <ítélet>Belső világ
   <ítélet>Cselekedetek/Akarat
    <ítélet>Kitartás/Következetesség
     <jelentés id="1">Ha belefogtál valamibe, folytasd is!</jelentés>
    </ítélet>
    <ítélet>Felelősség
     <jelentés id="2">Viseld tetted következményeit!</jelentés>
    </ítélet>
   </ítélet>
  </ítélet>
  <ítélet>Külső világ
   <ítélet>Hétköznapi élet
    <ítélet>Munka
     <jelentés id="3">A dologtalan, lusta ember az ételt sem érdemli meg
     </jelentés>
    </ítélet>
   </ítélet>
  </ítélet>
 </ítélet>
 <ítélet>Környezet
  <ítélet>Időjárás
   <jelentés id="4">A májusi eső hasznos.</jelentés>
  </ítélet>
 </ítélet>
</ítélet>
```

Megfelelő lekérdező felülettel kiegészítve az XML teljes mértékben ellátja és kielégíti a második pontban bevezetett ismérveket. Ennek megvalósításához természetesen szükséges valamilyen XML-t ismerő szkriptnyelv (PHP vagy PERL), amely az adatbázist webszerveren keresztül összekapcsolja a külvilággal.

Így az adatfeltöltéshez a feltöltő személyeknek nem szükséges ismerniük az XML nyelvet, sem egyéb formai vagy programozási technikákat. A megjelenítő, lekérdező és adatbeviteli felület teljes egészében a HTML nyelv szabványaira épül, így tetszőleges böngészőprogrammal az említett műveletek könnyedén elvégezhetők. Emiatt bárki korlátozás nélkül kapcsolódni tud a projekthez a világhálón keresztül akár a három kiinduló nyelven túl esetleges további nyelvek bevonásával is.

7 Záró gondolatok

A jelenlegi adatbázis-projektem egy komoly hiányt kíván pótolni, amelyre eddig csak részleges megoldás létezik, nevezetesen ilyen a *The Matti Kuusi international type system of proverbs*[4]. Outi Lauhakangas dolgozta ki a Matti Kuusi által összeállított közmondás-gyűjtemény internetes változatát, amely kizárólag forrás-megjelölést és kategóriákba sorolást nyújt, valamint csak keresésre ad lehetőséget a kutatók számára. A bemutatott közmondás-adatbázis a nyelvi adatokkal (a konkrét esetben a parömiákkal) kapcsolatos teljes spektrumot igyekszik felölelni, és általánosan hozzáférhető, bővíthető korpuszként rendelkezésre áll a kutatók (első sorban nyelvészek és folkloristák) számára.

Hivatkozások

1. Bańczerowski Janusz (a): *A kognitív nyelvészet alapelvei.* In: Magyar Nyelvőr online http://www.c3.hu/ nyelvor/period/1231/123107.htm.
2. Bańczerowski Janusz (b): *A világ nyelvi képe mint a szemantikai kutatások tárgya.* In: Magyar Nyelv online: http://www.c3.hu/ magyarnyelv/99-2/banczer.html.
3. Dobrovol'skij, Dmitrij (2002): *Phraseologie als Datenbank.* In: Hartmann, Dietrich/Wirren, Jan (Hrsg.): Wer A sägt, muss auch B sägen. Beiträge zur Phraseologie und Sprichworforschung aus dem Westfällischen Arbeitskreis. Hohengehren: Schneider Verlag. pp. 429-432.
4. Lauhakangas, Outi: *The Matti Kuusi international type system of proverbs.* In: http://lauhakan.home.cern.ch/lauhakan/cerp.html
5. O. Nagy Gábor (1999): *Magyar szólások és közmondások.* Talentum Kiadó.
6. Schneider, Gerold (2000): *Korpuslinguistik I. Morphologieanalyse und Lexikonaufbau.* In: http://www.ifi.unizh.ch/CL/gschneid/LexMorphVorl/Lexikon08.Corpus1.html
7. Tóthne Litovkina, Anna (1996): *Parömiológiai felmérés Magyarországon (Milyen formában és változatban élnek a legismertebb közmondások, és mi határozza meg az ismeretüket?).* In: Magyar Nyelv 4. pp. 439-458 .
8. Zunker, Giesela/Rapp Reinhard (1994): *Maschinenlesbare deutsch- und englischsprachige Textkorpora.* In: Hallwachs, D.W./Stütz, I. (Hrsg.). Sprache - Sprechen - Handeln. Akten des 28. Linguistischen Kolloquiums, Graz 1993, Band 2. Tübingen: Niemeyer, pp. 341-348.
9. Влахов, Сергей: *Сопоставителен речник на пословици.* - София: Издателство ЕТО 1998
10. Икономов, Николай: *Балканска народна мъдрост.* - София: Издателство на Българската академия на науките 1968

Népi Hiedelem Gyűjtemény Analízise Fuzzy Pseudo-tezaurusszal

Szaszkó Sándor [I]. Kóczy T László [I, II] Gedeon Tamás [III]

[I] Budapesti Műszaki és Gazdaságtudományi Egyetem
Távközlési és Médiainformatikai Tanszék
1117 Budapest, Magyar tudósok krt. 2.
{szaszko, koczy}@tmit.bme.hu

[II] Széchenyi István Egyetem
Villamosmérnöki és Informatikai Intézet
9026 Győr, Egyetem tér 1.
koczy@sze.hu [II]

[III] Ausztráliai Nemzeti Egyetem
Informatikai Tanszék
ACT 0200 Canberra
Ausztrália
tom.gedeon@anu.edu.au

Kivonat: Az ideális tezaurusz szócsoportokból áll. A csoport szavai egy olyan fogalomhoz tartoznak, amely absztrakt is lehet, vagyis a valóságban nem megtalálható. Kutatásunk célja automatikus tezauruszgenerátor módszer kifejlesztése és hangolása szavak dokumentumokban való közös-előfordulása alapján. A pszeudó-tezaurusz szócsoportjai egy-egy témát írnak le. E témák köré felépíthető a dokumentumok tartalmi csoportosítása. 2704 magyar népi hiedelem szöveg feldolgozását végeztük el. A lehető legnagyobb számú fogalom megtalálása céljából egy tapasztalati súlyt vezettünk be, melynek kiválasztását részletesen indokoljuk. Már a kutatás eddigi részeredményei is segítettek néprajzkutatóknak elemezni és megérteni a korpusz rejtett struktúráját.

Bevezetés

A természetes nyelvek sok hasonló szót használnak egy vagy több hasonló fogalom kifejezésére. Speciális szótárak használata válik szükségessé, ha szeretnénk az összes olyan dokumentumot visszakeresni, amelyek egy adott témához tartoznak. Tezaurusz kifejezések gyűjteménye, amelyben az azonos csoportba besorolt szavak egy adott fogalmat írnak le. A tezaurusz használatával felfedhetjük a kapcsolatot olyan dokumentumok között is, amelyek nem tartalmaznak azonos kifejezéseket, bár azonos témáról szólnak.

Az automatikus kulcsszó keresés a legelterjedtebb megoldás a dokumentumkeresésre, habár könnyen belátható, hogy kulcsszót nem tartalmazó dokumentum is lehet releváns a keresésben. Vegyük például azt az esetet, amikor a „puha számítástudo-

mány" (soft computing) kulcsszóra keresünk; a Fuzzy rendszerekről vagy neurális hálózatokról szóló szövegek nem lesznek benne a keresés eredményhalmazában, habár relevánsak a keresett témához. Ugyancsak nem találjuk ezzel a módszerrel azon közösségek dokumentumait, akik „számítástudományi intelligencia" (Computational Intelligence) kifejezést preferálják azonos kontextusban.

Az [1][4] tanulmányokban szavak hierarchikus közös-előfordulási mértékét javasoltuk az egyes szavak, illetve szócsoportok fontosságának jelzésére. A dokumentumok címében, alcímében, kivonatában szereplő szavak kiemelt fontosságot kaptak és hozzákapcsolódtak a dokumentumtörzs minden szavához. Általában is elmondható, hogy fuzzy logika alkalmazása automatikus dokumentumok keresésben nem új keletű, a legfontosabb eredmények a [5]-ban vannak összegezve.

A hiedelem gyűjtemény fuzzy élőfeldolgozása

Ha adott területről származó szöveget analizálunk, akkor a szavakat négy fő halmazba csoportosíthatjuk, ahogy ez az 1. ábrán is látszik.

A stop szavak a nyelv olyan eszközei melyeket szinte minden szövegben megtalálhatunk, a szöveg tartalmáról nem hordoznak információt. A relatív stop szavaknak hasonló szerepe van a stop szavakhoz, de csak adott típusú szövegek esetén, mint például a jogi dokumentumok esetén a „törvény" szó. Az általunk feldolgozott hiedelem gyűjteményben nem azonosítottunk relatív stop szavakat.

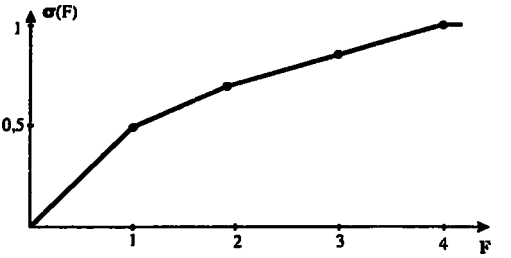
1. ábra Szókategóriák a dokumentumokban

A stop szavak eltávolítása után fennmaradó szavak a fontos szavak, melyek felhasználhatóak további analízisre. A fontos szavak egy része pontosabb információt ad a dokumentum tartalmáról, ezeket a szavalat nevezzük kulcsszavaknak.

A gazdag magyar néphagyományból 2704 db népi hiedelemszöveget bocsátott digitális formában rendelkezésünkre a a Néprajzi Múzeum. Ezen szövegek átlagos hossza 2–5 sor, vagyis igen rövid. Igen sok a régies, vagy egy-egy nyelvjárás szerinti szó, ezért szükség volt egy előfeldolgozási szótárra, amelyben kb. 13 000 szó (ragozott formákat is beleértve) 1704 fő bejegyzés alá lett besorolva. Ez után a korpuszt a K' mátrixban ábrázoltuk; a K' mátrix felső élén a szavakat soroltuk fel, míg oldalán a dokumentumokat.

Kritikus pont a K' mátrix előfordulási értékeinek fuzzy tagsági értékekké transzformálása. A tagsági érték fejezi ki, milyen mértékben jellemző egy szó az adott dokumentumra.

A traszformációra $\sigma(F)$ sigmoid függvényt választottuk [2]. A „S" formájú $\sigma(F)$ függvény konkrét alakját az adott korpuszhoz kell illeszteni. A dokumetumok rövidsége miatt a mi estünkben már az egyszeri előfordulásnak is nagy jelentősége van, az esetek 99,9%-ban nem fordul elő egy szó 4-nél többször egy dokumentumban. Ezek alapján alakítottuk ki a 2. ábrán látható függvény.



2. ábra A korpuszon használt sigmoid függvény

2.1 Szógyakorisági mérték

Későbbi használatra definiáljuk:

$$I_w = \sum_{i=1}^{N} \sigma_{d_i W} \tag{1}$$

Az l_w szógyakorisági mérték mutatja meg egy a w szó fontosságát a dokumentumgyűjteményben.

Szavak együttes-előfordulása alapján Fuzzy Pseudo-tezaurusz

Az ideális tezaurusz úgy definiálható, hogy minden szót hozzá rendelünk egy vagy több fogalomhoz. A fogalmak lehetnek absztraktak is, a való világban nem megtalálhatók. Két szót akkor tekintünk egymás szinonimájának, ha azonos fogalmakhoz tartoznak. Fuzzy tezaurusz esetén a fogalomhoz tartozás és így a szinonimaság mértéke (azaz az összetartozás foka a tezauruszban) fuzzy mérték, nulla és egy közötti szám.

A javasolt automatikus tezauruszgenerálás estén a fogalmak szerepét a dokumentumok töltik be, tehát akkor mondjuk két szóról, hogy szinonimák, ha azonos dokumentumokban szerepelnek. Természetesen ez a kapcsolat is leírható fuzzy mértékkel.

Sajnos az általános fogalmak és a dokumentumok között koránt sem létezik egy-egy leképezés. Egy dokumentum általában több absztrakt fogalmat is tartalmaz és sokat érintőlegesen taglal, így az automata módon generált tezaurusz nagy eltéréseket mutathat az ideálistól, ezért e módszerrel kapott tezauruszt pseudo- vagyis áltezaurusznak nevezzük.

Fuzzy pseudo-tezaurusz létrehozása:

1. lépés: Közös előfordulási mérték számítása

A számítások alapját a szöveg előfeldolgozásakor $\sigma(F)$ értékek adják, amelyek azt mutatják meg, hogy egy adott szó (W_i) milyen mértékben jellemző egy dokumentumra (D).

$$\mu'_{ij}(D) = \min(\sigma_{W_iD}, \sigma_{W_jD}) \tag{2}$$

$$\mu_{ij} = \frac{1}{C}\frac{1}{s}\sum_{z=1}^{N}\mu'_{ij}(D_z)$$

ahol C konstans feladata μ_{ij} értékét a [0,1] intervallumban tartani. C állandó az egész korpuszon, míg az s súly paraméter értéke változhat a szavak (i,j) függvényében is. A legegyszerűbb választás C értékére N, a dokumentumok száma, de ekkor az összes μ_{ij} érték nagyon kicsi. μ_{ij} jobban kihasználja a [0,1] intervallumot ha

$$C = \max_{i,j}\left(\frac{1}{s}\sum_{z=1}^{N}\mu'_{ij}(D_z)\right) \tag{3}$$

2. lépés: α-vágat

Ha a fontos szavak száma M, akkor a közöselőfordulási mértékek (μ_{ij}) egy MxM mátrixot alkotnak, hívjuk ezt W-nek. W mátrix mindkét oldalán a fontos szavak szerepelnek.

Mivel $\mu_{ij} = \mu_{ji}$ W ábrázolható egy irányítatlan gráffal. Válasszunk egy olyan α-t, amellyel elkészítve az α-vágatot – azaz kinullázva az összes α-nál kisebb értéket W-ben – csak 30-40 sor tartalmaz 0-tól különböző értéket. Ábrázoljuk ezt a redukált gráfot, ez reprezentálja a speudo-tezauruszt.

3. lépés: Maximum kikkek keresése

Ha a gráfban két pont (szó) éllel van összekapcsolva, akkor ők egymás szinonímái (kiterjesztett értelemben). Ha szavak egy részhalmazában mindenki mindenkinek kapcsolatban áll, akkor ők egy egy általánosabb értelemben vett fogalomhoz tartoznak.

A legnagyobb teljesen összekötött szócsoportok, maximum klikkek megkeresésével a korpusz főbb fogalmait azonosítjuk.

4. lépés: Fuzzy klikk

Sok esetben a maximum klikkeknek sok közös pontjuk van és csak egy-két szóban térnek el egymástól. Mivel az α-t önkényesen választottuk, ezért értelmes lehet megvizsgálni, hogy ezek az egymáshoz közeli klikkek azonos fogalmakat írnak-e le.
Válasszunk ki két klikket, melyek csak egy pontban különböznek és vizsgáljuk meg, hogy a két pont össze van-e kötve $\alpha'=0.7\alpha$ vágat szintjén. Ha igen, akkor a két klikket egyesítjük.

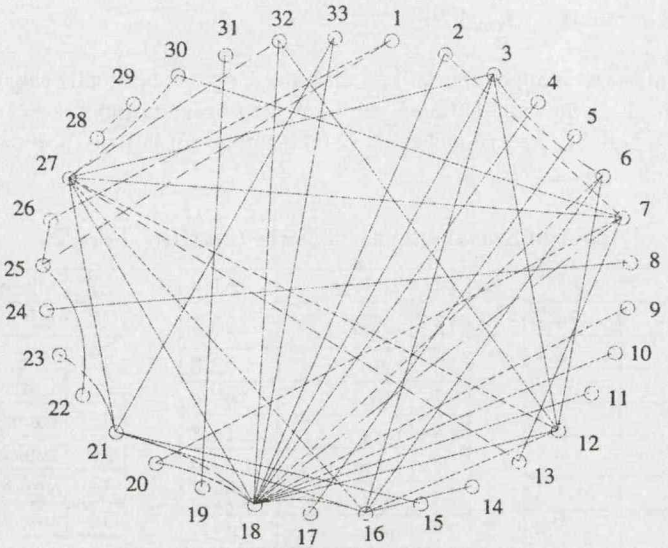
3.1 Súly s=1

A legtöbb fogalom felderítése céljából különböző s súly értékeket próbálunk ki.
Itt a közös előfordulás mértéke egyenesen arányos a szópárok együtt előfordulásának számával. Az I_w oszlop tanulsága szerint csak nagy gyakoriságú szavak maradtak benne az α-vágatban. A leggyakoribb szavaknak (mint pl. megy, tesz) van a legtöbb élük.

3. Táblázat Szavak listája súly s=1

NR	HUN.	I_w
1	ad	74.7
2	asszony	92.1
3	este	103.3
4	fent	69.1
5	fog	123.5
6	férj	66.1
7	gyermek	160.8
8	György	31.8
9	haza	42.8
10	3	83.7
11	ház	173.6
12	karácsony	90.4
13	kicsi	110.9
14	legény	41.7
15	Luca	52.8
16	lány	142.4
17	meghal	90.8
18	megy	218.4
19	mise	27.6
20	mond	108.3
21	Nap	172.3
22	ront	48.7
23	sok	86.0
24	szent	37.8
25	tehén	97.1
26	tej	61.3
27	tesz	177.8
28	tojik	32.3
29	tyúk	77.4
30	víz	98.9
31	éjfél	40.5
32	éjjel	87.2
33	év	94.7

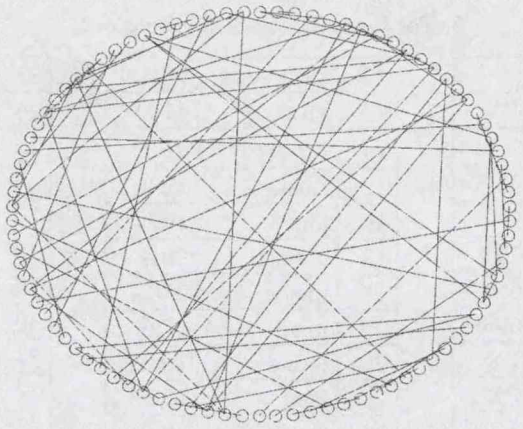
Habár nem is az összes, de sok szópár mutat jelentésbeli kapcsolatot. Találhatunk mögöttes logikát a maximum klikkekben, de általában egy-egy nagyon gyakori szó megjelenik, mint kakukktojás a csoportban. (Lásd 3.ábra)



3. ábra Pseudo-tezaurusz gráfja, súly s=1

3.2 Súly s=max(I_{WA} ,I_{WB})

A gyakori szavak dominanciájának elkerülésére osszuk el a közöselőfordulási mértéket, μ_{ij}-t a gyakoribb szó szógyakorisági mértékével (I_w). Ebben az esetben C=1 mivel I_w sohasem kisebb, mint μ'_{ij}.



4. ábra Pseudo-tezaurusz gráfja, súly s= max(I_{WA} ,I_{WB})

A legkisebb nem üres α-vágat 90 pontot tartalmaz. Minden bent maradó szó esetén I_w=0.5, tehát ezek a szavak csak egyetlen egyszer fordulnak elő az egész dokumentum gyűjteményben. Emiatt a feltárt kapcsolatoknak nem lehet értelmük, és ha megvizsgáljuk nincs is.

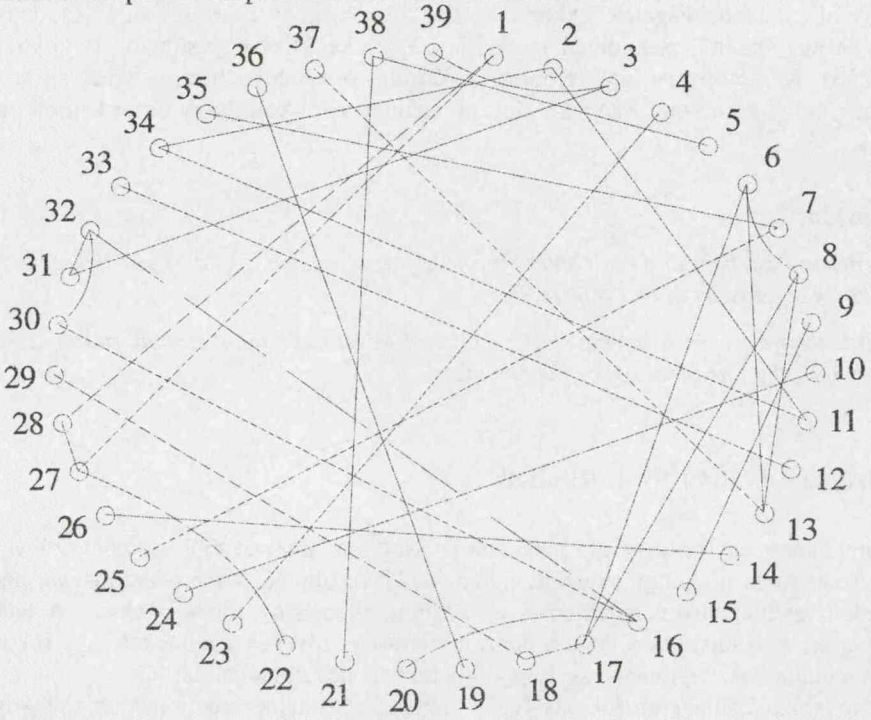
3.3 Súly s=1+ (max(I_{WA} ,I_{WB}))/20

Az előző két alfejezet alapján arra kell gondoljunk, hogy az optimális súly paraméter 1 és max(I_{WA}, I_{WB}) között van. Több súlytényező megvizsgálta után az s=1+ (max(I_{WA}, I_{WB}))/20 adta a legjobb eredményeket. A 2. táblázatban jól látható, hogy az I_w értékek széles skálán mozognak.

4. táblázat Szavak listája, súly s=1+ (max(I_{WA}, I_{WB}))/20

NR	HUN.	I_w
1	ad	74.7
2	bal	28.4
3	csibe	35.2
4	csörög	5.5
5	cédula	9.5
6	este	103.3
7	fecske	16.8
8	férj	66.1
9	gyermek	160.8
10	György	31.8
11	jobb	30.9
12	jön	91.1
13	karácsony	90.4
14	kenyér	51.4
15	kicsi	110.9
16	Luca	52.8
17	lány	142.4
18	megy	218.4
19	mise	27.6
20	nap	172.3
21	név	43.3
22	ront	48.7
23	szarka	6.7
24	szent	37.8
25	szeplő	10.3
26	süt	33.6
27	tehén	97.1
28	tej	61.3
29	tesz	177.8
30	tojik	32.3
31	tojás	43.8
32	tyúk	77.4
33	vendég	29.8
34	viszket	25.5
35	vér	17.6
36	éjfél	40.5
37	éjjel	87.2
38	ír	20.7
39	ültet	21.3

Az 5. ábrán látható gráfon az egyik pontnak sincs túl domináns, az eredményeket torzító szerepe, még az olyan kifejezetten gyakori szavaknak is, mint tesz, vagy megy nincs több, mint kettő éle, tehát sikerült elkerülni, hogy legyen néhány szó amely szinte minden más ponthoz kapcsolódik a 3. ábrához hasonlóan.



5. ábra Pseudo-tezaurusz gráfja, súly s=1+ (max(I_{WA}, I_{WB}))/20

5. táblázat Maximum klikkek

ad (1)	tehén (27)	tej (28)
bal (2)	jobb (11)	viszket (34)
este *(6)*	férj *(8)*	karácsony *(13)*
este *(6)*	férj *(8)*	lány *(17)*
Luca (16)	tojik (30)	tyúk (32)

csibe	ültet
csörög	szarka
cédula	ír
Fecske	szeplô
Fecske	vér
gyermek	kicsi
György	szent
jön	vendég
karácsony	éjjel

kenyér	süt
luca	Nap
lány	megy
megy	tesz
mise	éjfél
név	ír
ront	tehén
tojás	tyúk

A 3. táblázat az 5. ábra gráfjának maximum klikkjeit sorolja fel. Ezek a szócsoportok egy-egy általános értelemben vett, a korpusz számára fontos fogalmat határoznak meg. A 3. táblázat harmadik és negyedik sora csak egy szóban tér el. Ha megvizsgáljuk W mátrix alacsonyabb vágatait, akkor megállapíthatjuk, hogy a „Karácsony" és „lány" $\alpha'=0.8\alpha$ mértékben kapcsolódnak egymáshoz, így a két klikk egyesíthető. Az új klikk: este, férj, Karácsony és lány. Könnyű elképzelni olyan hiedelmeket, amelyek arról szólnak, hogy karácsony estéjén a lánynak valamit tenni kell, hogy férjet kapjon magának.

Két példa:

"Karácsony estélyén doboskát sütnek s a leány az elsővel kiszalad és amely legénnyel találkozik legelőször az lesz a férje."

"Karácsony estélyén a lánynak egy öl fát kell felvenni és ha a számuk páros, akkor férjhez meg, ha páratlan, akkor nem megy."

Összegzés és további kutatások

Automatikus tezauruszgeneráló metódust javasoltunk, amelyet több súlytényezővel is megvizsgáltunk, majd egy hatékony paramétert javasoltunk. A metódust magyar népi hiedelem gyűjteményen alkalmazva azonosítottunk néhány főbb fogalmat. A talált szócsoportok értelmesek voltak, a hozzájuk tartozó szövegek egymással nagy rokonságot mutattak. Az eredményeket népraj kutatók alkalmazhatónak tartják.

A továbbiakban szeretnénk egy fuzzy mértéket definiálni, amely megadja egy-egy talált szócsoport, vagyis „fogalom" és dokumentum közelségét, majd ennek segítségével klaszetrezzük a dokumentumokat.

Köszönetnyilvánítás

Köszönjük Darányi Sándornak és Kiss Ferencnek a segítségét, és hogy lehetővé tették a korpuszhoz, illetve az előfeldolgozó szótárhoz való hozzáférést.

Referenciák

[1] K. Chakrabarty, L.T. Kóczy, T.D. Gedeon, Analysis of fuzzy relational charts in information retrieval, IETR99-01, School of Computer Science and Engineering, University of New South Wales, Sydney, 1999.

[2] L.T. Kóczy, Interactive σ-algebras and fuzzy objects of type N, J. Cybernet. 8 (1978) 273–290

[3] L T. Kóczy, T. D. Gedeon and J. A. Kóczy, Fuzzy tolerance relations and relational maps applied to information retrieval, Fuzzy Sets and Systems 126 (2002) 49–61

[4] L.T. Kóczy, T.D. Gedeon, Information retrieval by fuzzy relations and hierarchical co-occurrence, Part II, TR97-03, Department of Information Engineering, School of Computer Science and Engineering, University of New South Wales, Sydney, 1997.

[5] S. Miyamoto, Fuzzy Sets in Information Retrieval and Cluster Analysis, Kluwer, Dordrecht, 1990, 259p

A szupermorféma

Nyelvtechnológia és szöveg

Kis Ádám
SZAK Kiadó Kft.
adam.kis@szak.hu

Kis Balázs
MorphoLogic Kft.
kis@morphologic.hu

Kivonat. Az előadás a szöveg, illetve meghatározott részeinek, szintjeinek körülhatárolásával, ennek számítógépes alkalmazásaival, illetve szintjeivel foglalkozik. Nem a számítógépes nyelvészetben hagyományosnak tekinthető nyelvtani, illetve tartalomelemzési szempontokat használja fel, hanem a szöveg mint komplex egység tartalmát vizsgálja. A három alkalmazási példa – a hivatkozási példa, a keresés és a fordítás – a szövegek tartalmi vizsgálatának problémáját a szövegek gépi összehasonlításának, illetve a mintaillesztésnek a korlátaira vezeti vissza: fő kérdése, hogy ezek a korlátok meghaladhatók-e, s lehet-e szövegeket tartalmuk alapján összehasonlítani, illetve keresni.

1. Bevezetés

A tudományos szövegmeghatározások a szöveg dimenzionális meghatározását lényegében teljesen szemantikai alapokon végzik. De Beugrande és Dressler meghatározásrendszere (de Beugrande-Dressler, 1923) 7 ismérvet sorol fel, amelyeknek teljesülniük kell minden „szövegszerűségen" ahhoz, hogy érvényes legyen rá a kommunikatív jelző, és ez elengedhetetlen feltétel, hogy ezt a nyelvi jelenséget szövegnek tekintsék. Mint írják, „...a nem kommunikatív szöveget nem tekintjük szövegnek".

A szövegtan rövid történelmében kialakult szövegmodellek (Van Dijk, Petőfi) mind olyan ismérvek alapján közelítik meg a szövegfogalmat, amelyek algoritmizálása minden, csak nem triviális (Tolcsvai, 2000).

A nyelvészeti szövegfogalom nagyon erősen kötődik a jelentéshez. „...a szöveg elsődleges rendeltetése valamilyen értelemreprezentáció" – írja Tolcsvai (2000).

A de Beugrande-Dressler-féle meghatározás sutasága egyenesen elvezet a számítógépes szövegkezelés problémájához: az nyilvánvaló, hogy a számítógépen a szöveg – karaktersorozat, de nem minden (nyelvi szempontból értelmezhető) karaktersorozat tekinthető szövegnek. Vajon léteznek-e számítógépes módszerek az emlegetett kétféle karaktersorozat megkülönböztetésére?

A szöveg a számítógépen mindenképpen egy karaktersorozat, amelyet nyelvi kódolással hoznak létre. Szöveggé valójában akkor válik, ha bár dekódolják. Egy karaktersorozat a számítógépen gyakorlatilag háromféleképpen határolható körül (lokalizálható, kereshető, jeleníthető meg):

1. Fájlként.
   Ez azt jelenti, hogy a sorozat valamennyi karakterét közös névtérben helyezik el, itt hozzárendelik egy egyértelmű és megismételhetetlen azonosítóhoz (ez a sorozatot tároló számítógépre vonatkozó korlátozás, ami viszont, az adott számítógép egyedi azonosítása folytán végeredményben kiterjeszti a teljes digitális térre).
2. A karaktersorozat valamely pontjának megadásával.
   - Az elterjedt szövegszerkesztők *egy* általános konvenciót tartalmaznak ebben a tekintetben: a **szó** határolását. Közismert, hogy a számítógépen a szó két **szóköz** közötti karaktersorozat. Ez a konvenció igencsak alkalmas például a grammatikai összefüggések vizsgálatára.
   - Ugyancsak megvannak a **mondat** elhatárolását végző kitüntetett karakterek (a mondatzáró írásjelek és a bekezdésjel), azonban ez figyelemmel a bekezdésjel tipográfiai funkcióira koránt sem egyértelmű (előfordul, hogy – tipográfiai meggondolás miatt – egy mondat akár 3 bekezdést is alkothat).
   - A szöveg létrehozója a szöveg tetszés szerinti pontját felszerelheti olyan jelöléssel, amely a későbbiekben megtalálható, és így azonosíthatóvá teszi a szövegrészt. Ilyen eszköz például a könyvjelző (bookmark).
   - Az előző pont kiterjesztésével szövegintervallumok kijelölésére is mód van, mégpedig oly módon, hogy a könyvjelző nem egyetlen szövegpontra hivatkozik, hanem egy összefüggő karaktersorozatra (melyet pl. a Word szövegszerkesztőben *kijelöléssel* fogunk egybe).
3. Minta segítségével. A karaktersorozat egy részét a rendszer úgy lokalizálja, hogy vizsgálja, megegyezik-e egy adott mintával.

A szövegnek tekintett karaktersorozatok körülhatárolása a számítógép alkalmazásának lényeges funkcióihoz tartozik hozzá. Ezek közül hármat elemzünk: a hivatkozást, a keresést és a fordítást.

2. A hivatkozás

A szövegek hivatkozásokkal való összekapcsolása egyáltalán nem a modern kor terméke. Ez az eljárás lényegében a lineáris olvasás kötöttségeinek feloldására szolgál. Voltaképpen az írás, illetve az írott szöveg közelítése a gondolkodás struktúrájához, amely sokkal inkább bonyolult hálóra emlékeztet, semmint szekvenciára. Ősi, nevezetes példa erre maga a Biblia, amely tulajdonképpen szabályos indexszekvenciális struktúrát alkalmaz a nemlineáris olvasás megadásához.

A Károli-Biblia ekképp kezdődik (Biblia, 1948)[4]:

> *A világ teremtése*
> *(v.ö. Zsolt 104)*
> *1. Kezdetben teremté Isten az eget és a földet.*
> ** rész 2.4.5. Zsolt 33, 6.8,9, 12, 135.5*

[4] Szent Biblia, az Istennek Ó és új Testamentumában foglaltatott Szent Írás.

Mint látjuk, a kétségtelenül alapvető nyitómondathoz egy sor helyhivatkozás tartozik, amely módot ad az olvasónak arra, hogy a szöveg szerkezet által sugallt sorrendtől eltérjen.

Így, a tartalomjegyzék segítségével ellapoz a Zsoltárok könyvéhez (az adott kiadás 493. oldalára, ott megkeresi a 33. részt (ebben segít, hogy az élőfejben megadják az adott oldalon levő rész számát), ott megkeresi a 6. szakaszt, ahol ezt olvassa:

6. Az úr szavára lettek az egek és szájának leheletére minden seregök.
** 1. Móz., 1,6.7*

Máris teljes a kereszthivatkozás, azonban ez technikai értelemben roppant nehézkes. Igazából csak azért alakulhatott ki, és azért használták, mert csak egyetlen (néhány) könyv volt az emberek birtokában, melyet igazából nem is olvastak, hanem „használtak". A Biblia hivatkozásai a pontos és következetes struktúrán, illetve jelölésrendszeren alapulnak. Ez a jelölésrendszer lényegében *metanyelvnek* tekinthető, hiszen következetesen áttevődik a jó fordításokba is, biztosítva azt, hogy a hivatkozásrendszer a különböző nyelvi változatokban egyaránt érvényesüljön.

Tekintsünk egy másik példát az irodalmi hivatkozásokra! A következő szöveg Umberto Eco egy tanulmányából vett facsimile részlet [Eco:1994]

jelzések gyakran igencsak félreérthetők. Carlo Collodi *Pinokkió*-ja így indul:

Kezdődik a mese: – Volt egyszer egy...
– Király! – szóltok közbe tüstént, kis olvasóim.
*Csakhogy, barátocskáim, ezúttal tévedtek. Nem királyról szól a mese. Hol volt, hol nem volt, volt egyszer egy darab fa.**

Roppant csavaros kezdés. Collodi először mintha azt jelezné, hogy mesébe fog kezdeni. S

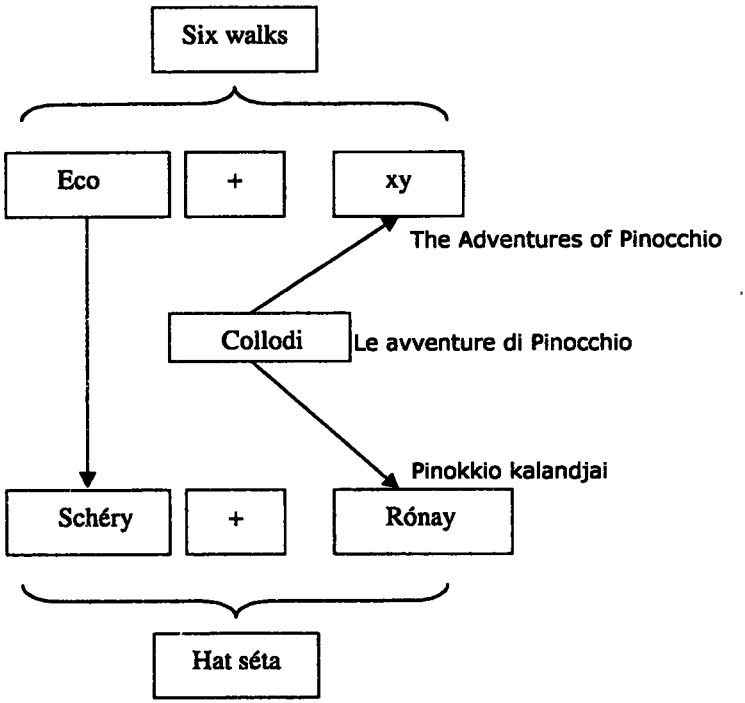
Mint látni fogjuk, ez a Bibilához képest igencsak bonyolult hivatkozásrendszer. A bonyodalmak azzal kezdődnek, hogy a beillesztett idézet – szemben azzal, hogy a bibliai szövegek végső soron ugyanannak a műnek különböző pontjaira hivatkoznak – egy egészen más műben őshonos. Míg a Bibliát kezelhetjük egyetlen strukturált szövegnek, addig Eco műve és Collodi *Pinokkiója* csak olyan alapon lehetnek egybefoglalható, ahogy minden ember rokon Ádámtól és Évától. A bonyodalmakat fokozza, hogy sem Eco tanulmánya, sem a Pinocchio, bár strukturált szövegek, nincsenek ellátva olyan könyvjelzőkkel, mint a Biblia. Mindehhez hozzájárul egy sajátos körülmény, mégpedig a szövegek nyelve.

A Biblia esetében egy-egy kötet rendszerint azonos nyelven szokott megjelenni, de mint említettük, a hivatkozásrendszer lehetővé teszi a különböző (korrekt) fordítások egységes kezelését. Esetünkben azonban – bár ez az első pillanatban nem érzékelhető, – a nyelvváltozatok figyelembevételével 5 szöveg is részt vesz a példában. Tételesen:

1. Eco angol nyelven írt szövege.
2. Collodi olasz nyelvű szövege.

3. Collodi szövegének angol fordítása[5].
4. Az Eco-szöveg magyar nyelvű fordítása.
5. Ezen belül a Collodi szöveg fordítása.

Nevezzük el a szövegeket a létrehozóikról (a fordításokat a fordítókról[6]), és vizsgáljuk meg a struktúrát!



Ezt a szövegkombinációt a mű valójában nem hivatkozásokkal oldja meg, hanem a megfelelő szövegek beemelésével. Az olvasó szempontjából ez tulajdonképpen kielégítő megoldás, hiszen a szövegek eredetiségi jellemzői inkább csak filológiai tekintetben érdekesek.

Van azonban a megoldásnak egy lényeges hátránya: az átemelt szöveg kikerül a szövegkörnyezetéből, és ezzel tulajdonképpen teljesen meg is változik. A Pinokkió néhány sora eredeti szövegkörnyezetében betöltött egyfajta funkciót, majd átkerült Eco szövegébe, és ott másfajta funkciót rendeltek hozzá. Figyeljük meg azonban, hogy ez a két funkció nem független egymástól. A Collodi-szövegrészletnek van egy olyan

[5] Nem volt alkalmunk meggyőződni arról, hogy vajon Eco a Pinocchio eredeti szövegét idézte, vagy annak angol fordítását, és ha az utóbbi történt, azt ki fordította angolra, Eco vagy más. Feltételezem azonban, hogy angol nyelvű volt az idézet, mert ellenkező esetben a hazai kiadói szokásnak megfelelően a fordításban is eredeti nyelven közölték volna, lábjegyzetben megadva a fordítást.

[6] A Collodi-szöveg angolra fordítóját XY-nak nevezzük (feltételezés szerint Nicolas Pewrella lehetett).

jelentése, amely lényegében szövegkörnyezettől függetlenül is létezik, értelmezhető. Ennek révén ez a szövegrészlet olyan viszonyba kerül az – otthagyott – környezetével, mint amilyen a mondatszintű nyelvstruktúrában a morfémák között szokásos. A szövegjelentés tekintetében morfémának tekinthetjük, és mivel szükségképpen a szó szoros értelmében vett morfémák halmaza, talán célszerű szupermorfémának nevezni.

Hozzá kell tenni, hogy a Pinokkió esetében ez a szövegrész azért válhatott szupermorfémává, mert maga a történet, és esetleg a szöveg is, közismert. Ugyanez fennáll a bibliai szövegeknél, ahol a hivatkozás mintegy figyelmezteti az olvasót (a szöveg „használóját"[7]), hogy az aktuális gondolatot kapcsolja össze egy másikkal, amelyet szintén ismer, illetve, ismerhet. A „szöveghasználatnak" más precedensei is vannak, pl. a jogszabályok olvasása. Nem véletlen, hogy a köznyelv az ilyen, mnemotechnikai jelentőségű kézikönyveket szívesen nevezi bibliának.

A számítógép a hivatkozások használatának relevánsan új lehetőségeit nyitja meg. Ha azonos szövegen belül akarunk hivatkozni, akkor módunkban áll a megfelelő hivatkozási pontról (melyet rendszerint a szövegekben eltérő színnel vagy aláhúzással, esetleg piktogrammal megjelölnek, de a folyamatos olvasás során jelöletlen is lehet, csak az egér ráhúzásával „bukkan elő") a szöveg egy másik pontjára ugrani. Ha nem egy szöveg (fájl) keretein belül vagyunk, akkor az úgynevezett hiperhivatkozások révén, az egész digitális kozmoszt átszelve juthatunk el vagy a hivatkozott fájlhoz (hogy ott más módszerrel keressük meg a szükséges helyet), vagy – az előző módszerrel kombinálva – annak egy pontjához.

Első pillanatra azt gondolhatnánk, hogy ez pusztán kényelmi szempont. Hiszen ugyanezt számítógép nélkül is el lehet érni: hogy megadjuk a hivatkozott helyet, oldalszámmal, illetve teljes bibliográfiával, és ebben az esetben a számítógép többlete „csak" annyi, hogy nem kell a megfelelő oldalra lapozni, illetve a könyvespolcon vagy könyvtárban keresgélni. A számítógép a hivatkozások mentén azonban többletet is nyújt: nemcsak a hivatkozott szöveget teszi elérhetővé, hanem annak környezetét is. Ez kegyetlen, de szükséges megoldás: megnehezíti a szövegek szabad értelmezésben való felhasználását, és megőrzi az eredetileg szándékolt jelentést.

3. Keresés

A számítógépes szövegkezelés másik kiemelkedő funkciója a keresés. Magát a funkciót itt nem ismertetjük, a Google-t vagy a Yahoo-t mindenki használja. Vegyük észre, hogy amikor szöveget keresünk a böngészők segítségével, lényegében ugyanaz történik, mint a hivatkozások esetében, csak éppen a tevékenységrendszer egy más pontján állunk. A hivatkozás során a hivatkozó szöveg szerzője kívánja a gondolatát valaki máséval kiegészíteni, és az olvasót ehhez a másik gondolathoz tereli. A keresés során magunk gondoljuk azt, hogy a gondolatunkat továbbviszi, illusztrálja vagy másféleképpen kiegészíti egy másik gondolat, és ezt próbáljuk megkeresni.

---

[7] A szöveg „használata" adott esetben azt jelenti, hogy a kognitivitást az emlékezet pótolja, azaz az olvasás adott esetben nem új ismeret szerzését, hanem valamely korábbi felidézését célozza. Természetesen ez sem zárja ki a kogníciót, mivel az ismeretek közötti új kapcsolatok új következtetések levonására alkalmasak. A hivatkozásokkal a szöveg linearitásán tudunk túllépni, a várt kontinuitás helyett vadonatúj kontiguumokkal bővítve világismeretünkt.

Ennek során két eljárásmenet van: az egyik az, hogy sejtjük, hol kereskedjünk. Emlékeink, információink vagy feltételezésein vannak arról, hogy az adott témával ki foglalkozott, és ekkor többé-kevésbé pontos adatokkal megkeressük az illető művet. A „civil" életben katalóguskutatással, könyvek átpörgetésével stb. A számítógépes világban fájlkereséssel, feltételezve, hogy a nekünk szükséges információt tartalmazó fájl a szerző nevével és/vagy a mű címével megtalálható.

Ez az egyszerűbb eset. Bonyolultabb az, amikor nem tudjuk pontosan a szerzőt, illetve a mű címét, és úgy próbálunk rátalálni a keresett szövegre. Ebben az esetben a keresőprogramok általában kombinációs segítséget nyújtanak, azonban itt találkozunk azzal a nehézséggel, hogy amíg az emberi agy képes bizonyos asszociációs íveket létrehozni egymástól formájuk tekintetében távol álló szövegelemek között (pl. a szinonimákat képes felcserélni), erre a számítógépnek kevesebb lehetősége van. Igazán sikert csak egzakt kereséssel lehet elérni: a keresési szándékot valószínűsítő keresés jobbára még a jövő zenéje.

Léteznek olyan megközelítések, amelyek megpróbálkoznak mind a tárolt szöveg, mind pedig a keresőkérdés tartalmának egyfajta ábrázolásával. (Vö. Prószéky, 2003). Ez rendkívül komplex feladat, és a jelenleg rendelkezésre álló számítógépes kapacitás mellett csak rendkívüli mértékben leszűkített tárgykörön belül alkalmazható – ugyanis a rendszerben szemantikai keretek felhasználásával létre kell hozni egyfajta világmodellt.

Egy másik lehetséges megoldás a keresendő és a keresett szöveg közötti kapcsolat ábrázolására másfajta, „közelítő" jellegű modellt felhasználni. Ekkor az absztrakciós szint sem a keresett, sem a keresett szöveg ábrázolása (inkább: transzformációja) esetén sem éri el a szemantikai szintet, azonban lexikális kapcsolatok megjelennek. Lexikális kapcsolatokat tezauruszok, illetve felszíni „ontológiák" (pl. WordNet) hoznak létre (Mihaltz, 2003). Ezeket a kapcsolatokat kell megjeleníteni és felhasználni a például teljes szövegű keresőrendszerben, amely így nemcsak a keresett kifejezés kulcsszavait, hanem az azokkal kapcsolatban álló szavakat, kifejezéseket tartalmazó szövegeket is megtalálja. (Vö. Prószéky-Kis, 1999, pp. 176-202.)

Nézzünk egy példát! Ha egy szövegben idézetet akarok szerepeltetni, általában emlékezetből beírom, azután megpróbálom a számítógép segítségével ellenőrizni, hogy helyesen idéztem-e. Ez a feladat nem igazán nehéz, ha pontosan emlékszem a szerzőre, a mű címére. Elég felütni a kötetet, és megtalálni. A gyakorlat azonban azt mutatja, hogy sokan szívesen idéznek, de a memóriájuk nem tökéletes. Milyen kellemetlen például egy ilyen eset (fiktív szöveg):

> *„... szépen fejezi ki ezt Radnóti: Világunknak elme kell nagy fénybe, mely igazodni magára mutat!"*

Ha a szerző biztos a dolgában, így hagyja, és az olvasók között biztosan lesz olyan, aki kapásból rájön, hogy ez nem Radnóti szövege, hanem József Attiláé. Ha ezt felismerjük, azért jó aggályosnak lenni, hátha így sem pontos az idézet. Beírjuk hát az egész szöveget a Google-ba. A válasz az, hogy ennek a keresésnek „egyetlen dokumentum sem felel meg." Ennek alapján arra kell következtetni, hogy ez a szöveg pontatlan, nem felel meg az eredetinek. A jelenlegi eszközrendszerben felhasználónak kell megfelelően „ravasznak" lenni ahhoz, hogy megfelelő választ kapjon. Olyan elemeit kell megkeresnie az idézetnek, amelyek egyrészről megfelelően szignifikánsak, azaz várhatóan elvezetnek az eredetihez, más részről kellően különösek ahhoz, hogy

belátható számú válasz keletkezzék. A nyelvérzék azt mondja, hogy ez a szócsoport: „elme kell nagy fénybe", megfelel ezeknek a kritériumoknak. A válasz meg is jön, az idézet helyesen *„társadalmunkba, elme kell, nagy fénybe', mely igazodni magára mutat*".

Mit kell ahhoz tenni, hogy ezt a ravaszkodást a gép elvégezze?

A számítógépnek képesnek kell lennie annak felismerésére, hogy két különböző szövegrész eléggé hasonlít egymáshoz. Mondhatjuk, hogy ez mindössze egy másik megfogalmazása annak a korábbi tételünknek, hogy a keresendő és a keresett szöveg közötti kapcsolatok ábrázolására van szükség. Megfelelő kapcsolat lehet például, ha a keresendő és a keresett szövegrész tartalmi (kulcs-) szavai – egy kis részük kivételével – megegyeznek vagy szinonimái egymásnak. A problémát azonban sokszor a szövegek matematikai jellegű összehasonlítására vezetik vissza, amely kizárja a szövegek nyelvi szerkezetének vagy tartalmának felhasználását, mégis sokszor alkalmas a hasonló szövegek közötti releváns kapcsolatok felfedezésére. Ez mellesleg bevett eszköz a fordítás számítógépes támogatásában.

4. Fordítás

Amikor a számítógéppel segített fordításról beszélünk, fontos hamar leszállni a fellegekről. A számítógépes fordítással kapcsolatban egy Alekszejev nevű professzor előadását hallgattam, aki elmondta, hogy a probléma 95%-ig meg van oldva, a fennmaradó 5% miatt azonban mindenképpen szükséges az emberi beavatkozás. Ez 1963-ban volt. Lehet, hogy Alekszejev túlzott a 95%-kal, azonban a megközelítés 40 év múltán sem jobb. Kétségtelen, hogy meghatározott igényszint kielégítésére alkalmas az ember nélküli fordítás, azonban a realitás mégis az, hogy a számítógép igazán hasznos két funkcióban lehet: vagy a szövegmegértés segítésében, minek során a cél igazából nem jól formált szöveg létrehozása, hanem az információ többé-kevésbé hiteles közvetítése nyelvi transzfer útján. A másik lehetőség az, hogy az egyszer megszületett emberi megoldások újbóli felhasználásának hozzáférhetővé tétele. A módszer lényege az, hogy amit egyszer lefordítottak, azt felesleges újra lefordítani, ha az eredeti szöveg valami miatt megismétlődik (és az a tapasztalat, hogy a szövegek mennyisége sokkal nagyobb, mint a szövegek fajtáinak száma, ami arra mutat, hogy az ismétlődés szükségszerű). Következésképp, ha egy forrásszöveget és egy fordítást együtt eltárolnak, akkor mód van arra, hogy egy újabb fordítási feladat során, megvizsgálva és megállapítva, hogy a forrásszöveg létezik a tárban, akkor annak újbóli fordítását nem kell elvégezni, elegendő elővenni a tárban hozzá asszociált tárgynyelvi szöveget. Ez a feladat voltaképpen nem különbözik a keresési feladattól. A haszna azonban mégis korlátozott, mert a forrásszöveg tekintetében gyakorlatilag 100%-os egyezés szükséges. Vajon lehet-e alkalmazni a módszert nem csak egyező, hanem hasonló szövegekre?

Erre a mai fordítástámogató programokban létezik megoldás, amely azonban tisztán matematikai, nincs nyelvészeti vonatkozása. A szöveget karakterkódok sorozatának tekinti, s ezeket a fuzzy logika elvei szerint hasonlítja össze. Jelenleg folynak kísérletek arra, hogy nyelvtani szerkezet, sőt, esetleg szemantikai tartalom alapján fedezzünk fel hasonlóságot (Kis-Lengyel, 2003; Gröbler-Hodász-Kis, 2004). A fordításban viszont nem ez jelenti az egyedüli problémát: a korábbi szövegek ugyanis fordításukkal együtt vannak tárolva, s amikor a forrásszöveghez „csak" hasonlót találunk, a

tárolt fordítás is legfeljebb hasonló lesz a kívánt fordításhoz. Így a gépi fordítástámogatásban az is kutatás tárgya, hogy a hasonlóság által reprezentált különbséget hogyan lehet csökkenteni, a tárolt fordítást hogyan lehet automatikusan átalakítani a tényleges forrásszövegnek megfelelően.

5. Összefoglalás

A szöveg számítógépes feldolgozása napjainkban nehéz feladatokat ró a mesterségesintelligencia-kutatókra. Neumann annak idején a számítógép lehetőségeit ekképp határozta meg

„...a tulajdonképpeni célkitűzést számokkal való műveletekkel kell először helyettesítenünk. Ez olyasmi, amit a gép maga nem tud elvégezni. Tehát előzetesen meg kell fontolni, hogy a kérdéses probléma hogyan fordítható le számokkal végzendő műveletekre." ([Neumann, 1954]:229).

A feladat világos: minden olyan szövegekkel kapcsolatos jelenséget, amelyet a szövegnyelvészet határozottan és mélyen szemantikai szinten határoz meg, le kell fordítani számokkal végzendő műveletekre. Ennyi. *Hic Rhodos, hic salta.*

Irodalom

BEUGRANDE, Robert De; DRESSLER, Wolfgang: Bevezetés a szövegnyelvészetbe. Corvina, É.n. [Beugrande-Dressler]

DIJK, Teun van: Some aspects of text grammmar. The Hauge: Mouton. in [Beugrande-Dressler] [Dijk]

ECO, Umberto (1994): Hat séta a fikció erdejében. Európa Könyvkiadó, Budapest. [Eco, 1994]

GRÓBLER, Tamás-HODÁSZ, Gábor-KIS, Balázs (2004): MetaMorpho TM: A Rule-based Translation Corpus. In: Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC-2004), Lisbon, Portugal.

KIS Balázs-LENGYEL István (2003): Új módszerek az emberi fordítás gépi támogatásá-ban. In: Az I. Magyar Számítógépes Nyelvészeti Konferencia gyűjteményes kötete, Szegedi Tudományegyetem, Szeged.

MIHÁLTZ Márton (2003): Magyar főnévi WordNet-ontológia létrehozása automatikus módszerekkel. MSZNY 2003, Szegedi Tudományegyetem, Szeged.

NEUMANN János (2003): A számítógép és az agy. In Neuman János Válogatott írások. TypoTeX, Budapest. [Neumann]

PETŐFI S. János (1990): Szöveg, szövegtan, műelemzés. Országos Pedagógiai Intézet, Budapest, 1990. [Petőfi, SZSZM]

PRÓSZÉKY Gábor (2003): NewsPro: automatikus információszerzés gazdasági rövidhírekből. MSZNY 2003, Szegedi Tudományegyetem, Szeged.

PRÓSZÉKY Gábor-Kis Balázs (1999): Számítógéppel emberi nyelven. SZAK Kiadó, Bicske.

Szent Biblia, az Istennek Ó és új Testamentumában foglaltatott Szent Írás. Brit és Külföldi Biblia Társulat és Magyar Biblia Társulat, Budapest, 1948. [Biblia, 1948]

TOLCSVAI Nagy Gábor (2001): A magyar nyelv szövegtana. Nemzeti Tankönyvkiadó, Budapest. [Tolcsvai, 2001]

VII. Pszichológiai szempontú szövegfeldolgozás

A LAS VERTICUM időmodulja

Ehmann Bea

MTA Pszichológiai Kutatóintézet,
1132 Budapest, Victor Hugó u. 18-22.
ehmannb@mtapi.hu

Kivonat. Az előadás a LAS VERTICUM szó feletti tartalomelemző programcsomag időmoduljait és kategóriáit, valamint az ezzel kapcsolatos pszichológiai validitási vizsgálatokat ismerteti. A szubjektív időélmény tartalomelemzéssel nyert eredményeit a konstruktum validálás során az El Meligi által javasolt pszichometriai konstruktumokkal, a kritérium validálás során pedig az Antonovsky-féle koherenciaérzék faktorokkal hasonlítottuk össze.

A LAS VERTICUM szó feletti tartalomelemző programot a Pécsi Tudományegyetem, az MTA Pszichológiai Kutatóintézete, a Morphologic Kft és a Gödöllői Szent István Egyetem közösen fejlesztette ki (László, 2004). Működésének elmélete a narratív pszichológia és a narratív pszichológiai tartalomelemzés (László, Ehmann, Péley, Pólya, 2002; László és Ehmann, 2003; László és Ehmann, 2004).

A szubjektív időélmény tartalomelemzéses vizsgálatának elméleti alapjai

A szubjektív időélmény vizsgálatainak pszichológiai hagyományairól korábban már több publikációban beszámoltunk. A pszichoanalitikus alapokról lásd Ehmann (2004c), a pszichometrikus és a tartalomelemzéses hagyományokról pedig lásd Ehmann (2004a és 2004b).

A LAS VERTICUM időmoduljai e két szoftverhez képest több újdonságot tartalmaznak. Miként a többi modulnál is, szerkezeti újdonság, hogy a program nem csak szavakra, hanem több tagból álló kifejezésekre is találatot ad, szemléletbeli újdonság pedig az időkategóriák finomabb felbontása (Ehmann, 2003).

A LAS VERTICUM program időmoduljai és kategóriái

IDŐHORG MODUL: Az időtengelyre ráhorgonyozható tematizációk

FIXHORG /Konkrét, kezdeti lehorgonyzás/ Példák: amikor, épp, azon a napon, stb.
IDŐKONT /Tartósság, folyamatosság/ Példák: egyfolytában, éveken át, stb.
IDŐPERF /Befejezettség/ Példák: addig a napig, utoljára, végül stb.

ISM /Ismétlődés/ Példák: gyakran, megint, szoktunk, újra, stb.

IDŐSZABAD MODUL: Az időtengelyre nem ráhorgonyozható tematizációk

ÖRÖKIDŐ Példák: mindig, állandóan, örökké, stb.
SOHAIDŐ Példák: soha, semmikor, stb.
BIZONYTALAN Példák: valamikor, bármikor, stb.

IDŐLÉPTÉK MODUL

NAPTÁR Példák: hétfő délután, két évvel ezelőtt, hajnalban, stb.
KORSZAK Példák: gyermekkoromban, vénségemre, stb.
ÜNNEPEK Példák: a születésnapom előtt, karácsonykor, stb.

Kísérleti stádiumban lévő kiskategóriák:

JELEN Példák: ma, mostanában, a mai napon, pillanatnyilag, stb.
JÖVŐ Példák: holnap, fognak, leszek, stb.
RÉGMÚLT Példák: annó, rég, régen, régebben, stb.
AZUTÁN Példák: aztán, azt követően, később, a következő héten, stb.
ÉSAKKOR Példák: 'és akkor'.

A LAS VERTICUM időmoduljainak validálása

Módszer

Szövegelemzés: A szubjektív időélmény tartalomelemzéses vizsgálatakor is a munkacsoportunk által használt normál rétegzett mintával dolgoztunk. Összesen 83 vizsgálati személyt beszéltettünk a legkülönfélébb önéletrajzi témákról, fontos életeseményekről /a részletekről lásd László, 2004/. A teljes interjús szöveganyagon /281.306 szó, szóköz nélkül számolva 1.467.858 karakter/ futtattuk le a LAS VERTICUM program LIN-TAG nevű szövegelemző összetevőjét az ATLAS.ti program által kínált felhasználói felületen. Az egyes időkategóriákra összesen 12.323 találatot kaptunk. Ezt a gyakorisági táblázatot vittük át SPSS programcsomagba.

Statisztikai feldolgozás: Az időkategóriák LIN-TAG-gal kapott gyakorisági értékeinek alsó és felső kvartiliseit kétmintás t-próbával hasonlítottuk össze az EL Meligi és az Antonovsky kérdőív adataival.

Eredmények

I. Konstruktum validálás: A szubjektív időélmény tartalomelemzéssel nyert eredményeinek összehasonlítása az El Meligi (1972) által javasolt pszichometriai konstruktumokkal:

1. El Meligi időáramlás élmény skála:
Szignifikáns eredmények:
Az IDŐHORG modul IDŐKONT skálája ($P<0.01$)
Az IDŐSZABAD modul ÖRÖKIDŐ skálája ($P<0.04$)
Értelmezés: Az az eredmény, hogy a minta felső kvartilise szignifikánsan többször tematizálta az idő folyamatosságát jelző utalásokat, illetve hogy állításait az 'örökidőbe' távolította /pl. 'mindig azt mondta'/, összhangban van az EL Meligi skálával. A szerző szerint ez a kategória patológiás esetekben tartalmazza a tudatosság szintjének disszociatív tendenciáit vagy fluktuációit is. Elképzelhető, hogy az idő folyamatosságának túlzott előtérbe helyezése az élmények naptári idő szerinti tagoltság zavarával jár együtt.

2. El Meligi idői orientáció skála:
Szignifikáns eredmények:
Az IDŐSZABAD modul SOHAIDŐ ($P<0.05$) és ÖRÖKIDŐ ($P<0.05$) skálái
Értelmezés: Ezt az eredményt is nagyon jelentősnek tartjuk, annyiban, hogy El Meligi szerint ez a faktor arra utal, hogy a patológiás személy nehezen észleli, hogy valamely esemény a jelenhez, a múlthoz vagy a jövőhöz tartozik-e, illetve nehezen képes az eseményeket az idői szekvenciában elhelyezni. Ha tehát valaki a hétköznapi beszédében fokozottan tematizálja a 'soha' és a 'mindig' típusú kifejezéseket, az azt jelzi, hogy a gondolatáramlás kevéssé horgonyozódik le a naptári időtengelyhez.

3. El Meligi tapasztalati életkor skála:
A már elkészült LIN-TAG modulokkal nem kaptunk szignifikáns eredményeket. Azonban a még kísérleti stádiumban lévő RÉGMÚLT időkategória $P<0.03$ szinten szignifikáns volt ezzel a konstruktummal. Azaz azok a személyek, akik a tényleges koruknál fiatalabbnak vagy öregebbnek érzik magukat, több olyan típusú kifejezést használnak mint pl. annó, rég, régen, régebben, stb.

II. Kritérium validálás: A szubjektív időélmény tartalomelemzéssel nyert eredményeinek összehasonlítása az Antonovsky-féle koherenciaérzék kérdőívvel (Antonovsky, 1987)

1. 'A világ kezelhetősége' /manageability/ faktor:
Szignifikáns eredmények:
Az IDŐHORG modul FIXHORG skálája ($P<0.03$)

Értelmezés: A világ kezelhetősége összefügg azzal, hogy képesek vagyunk az élet eseményeit a naptári időtengelyhez lehorgonyozni. Akik ebben jobbak, azok több konkrét idővel jelölt kifejezést használnak a mindennapi beszédjükben. Két kísérleti stádiumban lévő modul /AZUTÁN, ÉSAKKOR/ megerősíteni látszik ezt a jelenséget: akik jobban kezelik a világot, azok még a traumatikus eseményeket is igyekeznek lineáris időszekvenciában elhelyezni.

2. 'A világ jelentéstelisége' /meaningfulness/ faktor:
Az IDŐHORG modul IDŐKONT /folyamatosság/ skálája szignifikáns ($P<0.02$), az ISM /ismétlődés/ skálája tendenciaszerű ($P<0.08$) összefüggést mutatott. A kísérleti kategóriák közül a JELEN és a BEFEJEZETT JELEN mutattak összefüggést e faktorral, vagyis a világot jelentéstelinek megélő személyek beszédjében több olyan kifejezés szerepel, mint ma, mostanában, attól fogva, azóta, stb.

Hivatkozások

Antonovsky, A. (1987): *Unraveling The Mystery of Health.* Jossey-Bass, San Francisco.

Ehmann B. (2003): A LAS VERTICUM narratív pszichológiai tartalomelemző rendszer időmodulja. In: *Magyar Számítógépes Nyelvészeti Konferencia, MSZNY 2003, Szeged. Absztrakt Kötet.* Szegedi Tudományegyetem Informatikai Tanszékcsoport, Szeged. 288-290-291.

Ehmann B. (2004a): Elbeszélt élettörténeti epizódok időstruktúrája. In: László J., Kállai J., Bereczkei T. (Szerk.): *A reprezentáció szintjei.* Gondolat Kiadó, Budapest, 339-354.

Ehmann B. (2004b): tartalomelemzési módszerek a szubjektív időélmény vizsgálatára laikus beszélők szövegeiben. *Magyar Pszichológiai Szemle*, 3.

Ehmann B. (2004c): A szubjektív időélmény mintázatainak pszichoanalitikus és narratív pszichológiai párhuzamai. *Pszichológia*, 24, 4, 403-425

El-Meligi, A. Moneim (1972): A technique for exploring time experiences in mental disorders. In: Yaker, H., Osmond, H. and Cheek, F. (Eds.): *The Future of Time.* The Hogarth Press, London 220-272

László J. (2004): *Morfológiai-lexikai szint feletti pszichológiai tartalomelemző programok fejlesztése. NKFP Projekt Kutatási beszámoló.* Oktatási Minisztérium, Kézirat

László J., Ehmann B. (2003): LAS Verticum: Egy szó feletti tartalomelemző szoftver. In: *Magyar Számítógépes Nyelvészeti Konferencia. Szeged, december 10-11. Absztrakt Kötet.* Szegedi Tudományegyetem Informatikai Tanszékcsoport, Szeged. 288-289.

László J., Ehmann B. (2004): Narratív pszichológia és narratív pszichológiai tartalomelemzés. *Magyar Pszichológiai Szemle*, 3.

László, J., Ehmann, B., Péley, B., Pólya, T. (2002): Narrative psychology and narrative psychological content analysis. In: László, J., Stainton Rogers, W. (eds.): *Narrative Approaches in Social Psychology.* Budapest, New Mandate, 9-25.

A LAS Verticum tagadás és self-referencia modulja

Hargitai Rita

Pécsi Tudományegyetem, Bölcsészettudományi Kar, Pszichológiai Intézet
7624 Pécs, Ifjúság útja 6. hargitairita@freemail.hu

Kivonat: A LAS Verticumot három összetevő alkotja: a LINTAG elnevezésű kódoló program, az ATLAT.Ti fogalmi hálózatépítő program és az SPSS statisztikai programcsomag. A Morphologic Kft-vel közösen kidolgozott LINTAG nevű nyelvi elemző program tagadás és self-referencia modulja az adott jelenség szempontjából pszichológiai relevanciával bíró szókategóriák tagjainak előfordulási gyakoriságát jelzi. A tanulmány a tagadás és a self-referencia nyelvi markerek szintjén történő operacionalizálási folyamatát, valamint a modulok megbízhatóságára és érvényességére vonatkozó empirikus kutatási eredményeket mutatja be.

1 A tagadás pszichológiai jelentése

A tagadás fogalmának értelmezése a pszichológiai elméleteken belül nem mutat egységes képet: Freud (1925) szerint a tagadás az ítéletalkotás intellektuális, tudatos funkciója, míg mások (Szondi, 1956) amellett érvelnek, hogy a tagadás az én legáltalánosabb, legemberibb és – bizonyos esetekben – legfatálisabb *állásfoglalása*. E „nemet mondás" formája és különösen mértéke gyakran meghatározza az egyén és a közösség egymáshoz való viszonyát: a negációs funkció biztosítja az egészséges ember környezetéhez és morális standardokhoz való alkalmazkodását, ez garantálja a társadalom működőképességét bizonyos szükségletek, vágyak és képzetek elfojtása, gátlása révén. A „nemet mondás" azonban szélsőséges mértékben is megjelenhet: a tagadás túlsúlya mögött a világ értéktelenítése áll, amely minden esetben valamilyen énes veszélyre, destrukcióra – például a preszuicidális szindróma beszűkülésére (Kézdi, 1995), illetve az alkoholizmus vagy a narkománia következtében fellépő autoagresszióra – utal. Egy adott élettörténeti narratívumban felbukkanó, tagadásra utaló nyelvi markerek mennyiségi paramétereinek meghatározása így több szempontból is indokolt.

1.1 A tagadás modul operacionalizálása és reliabilitása

A LAS Verticum – Morphologic Kft-vel közösen kidolgozott – LINTAG nevű kódoló programja a tagadás szempontjából pszichológiai relevanciával bíró szókategóriák tagjainak előfordulási gyakoriságát jelzi. A tagadás lehet *explicit, vagy nyílt* (pl. nem, nincs, sem, stb.) és *implicit, vagy rejtett* (pl. felelőtlen, gondtalan), amelyre csak szó-

tani elemzésből, illetve a névszói kifejezések elemzéséből (pl. nélkül, helyett, kívül) lehet következtetni. A tagadás modul reliabilitására vonatkozó vizsgálat a modul általi gépi- és a kézi kódolás eredményének összehasonlításán alapult: az elemzett 10.000 szavas szövegbázist harminc különböző vizsgálati személytől származó egy-egy rövid – teljesítmény-, veszteség-, félelem-, rossz-, illetve jó tárgykapcsolati – történet alkotta. A kapott eredmények: helyes találat 99%, téves riasztás 1%, míg a kihagyások értéke 7‰. A fenti adatok tükrében a modul megbízható.

1.2 A tagadás modul validitására vonatkozó empirikus vizsgálatok

A tagadás modul validitásának tesztelése során a vizsgálati csoportot előzetes pszichiátriai múlttal nem rendelkező, egészséges személyek rétegzett mintája alkotta (N=73). A szövegelemzés alapját a fentiekben felsorolt, eltérő témájú élettörténeti narratívumok képezték. A vizsgálat során minden esetben a tagadást jelző nyelvi markerek előfordulásának relatív gyakorisági értékével számoltunk, ezáltal iktatva ki a történetek hosszának befolyásoló hatását.

1) Hipotézisünk szerint a negatív életesemények – félelem, szorongás, tárgyvesztés, kapcsolati kudarc – olyan élményfeldolgozást hívnak életre, amelyben a környezethez és a morális standardokhoz való alkalmazkodás nehezített, így a szükségletek és a képzetek elfojtása, gátlása, elidegenítése lesz jellemző. Mindez a tagadás megnövekedett mértékében ölthet testet. A fentiek alapján az élettörténeti narratívum témája, pontosabban annak élményminősége szerint különbséget várunk a tagadás mennyiségi vonatkozásában. Az eredmények igazolják elvárásunkat: a negatív életeseményeket megjelenítő narratívumokban a tagadás nyelvi formái szignifikánsan nagyobb számban vannak jelen (M_{Tagad}=0,038 vs. 0,027; t(71)=2,65, p<0,01).

2) A továbbiakban a veszteség-, illetve a félelem-/szorongástörténeteket használtuk szövegkorpuszként, s a LINTAG révén a tagadás modul vonatkozásában kapott gyakorisági értékek alsó és felső kvartiliseit kétmintás t-próbával hasonlítottuk össze egy projektív személyiségvizsgáló eljárás, az ún. Tematikus Appercepciós Teszt (TAT) (Murray, 1938) eredményeivel. Várakozásunk szerint a tagadó szerkezetek magasabb értéke azokat a személyeket jellemzi, akiknél kimutatható a funkcióbeszűkülés, a világ értéktelení-tése. A kapott eredmények tükrében mindez megerősítést nyert: a veszteségtörténetek esetén azokat a személyeket jellemzi a tagadó szerkezetek dominanciája, akiknél hiányzik az energizáltság, a kitartó, sikeres tevékenység, a vágyak tettekre váltása, azaz a teljesítmény iránti szükséglet (M_{Ntelj}=2,55 vs. 1,5; t(34)=2,491, p<0,05). Mindehhez társul a környezet részéről az a ráhatás, amely a hiányban manifesztálódik: a v.sz. nélkülöz valamit (családot, barátokat, státust), amire szüksége lenne a léthez, a boldogsághoz ($M_{Phiány}$=1,05 vs. 2,05; t(34)=-2,64, p<0,05). Inrapszichés szinten sajátos beszűkülés jön ezáltal létre, így a személynek nincsen lehetősége az örömre és az előbbre jutásra.

3) Vizsgálatunk harmadik fázisában ismét a fenti történeteket használtunk szövegkorpuszként és a TAT Press Veszteség konstrukcióját tekintettük csoportosító változónak, ahol e dimenzió alacsony, illetve magas értékeit kétmintás t-próba során hasonlítottuk össze bizonyos kérdőíves- (BDQ, BFQ, PLS) és projektív személyiségvizsgáló eljárások eredményeivel. Környezeti ráhatásként, azaz pressként a veszteség a hiányhoz hasonló állapotot jelent, azzal a különbséggel, hogy a veszteség a történet folya-

mán következik be. A pszichodinamikus interpretáció értelmében a depresszió, mint pszichopatológia a reális- és/vagy szimbolikus tárgyak elvesztésére, illetve a tőlük való megfosztottságra adott pszichés válasz (Freud, 1917). Így várható, hogy statisztikailag szignifikáns a kapcsolat a veszteségélmény és a Beck Depresszió Kérdőív pontszáma között (M_{BDQ}=3,14 vs. 5,27; t(34)=-2,062, p<0,05). Mindez együttjár a külső és a belső ingerek alacsonyabb mértékű kezelhetőségével (M_{PLSKez}=14,09 vs. 11,87; t(34)=2,735, p=0,01), valamint az elégedetlenség és a düh szabályozásának képességével (M_{BFQImp}=37,09 vs. 31,93; t(34)=2,082, p<0,05). Mi több, a sikeres megküzdés érdekében a v.sz. másoktól vár segítséget, védelmet és támogatást ($M_{Ntám}$=2,47 vs. 3,8; t(34)=-2,63, p<0,05). A veszteségélmény mentén szerveződő intrapszichés történések a szöveg nyelvi formáinak szintjén a tagadó szerkezetek megnövekedett arányában manifesztálódnak (M_{Tagad}=0,033 vs. 0,056; t(33)=-2,4, p<0,05). A fentiek alapján a tagadás azonosítására kidolgozott nyelvi modul reliabilitására és validitására vonatkozó vizsgálatok egyaránt igazolást nyertek.

2 A self-referencia pszichológiai jelentése

A modern pszichológiában kitüntetett helyet foglal el az énként való létezés problematikája. E rendkívül kiterjedt kérdéskör alapos tanulmányozása messze meghaladja jelen tanulmány terjedelmi korlátait, így az adott jelenséget leszűkítve csak utalnék a self-referencia szempontjából releváns pszichológiai fogalmakra: birtoklás és énszűkítés, létezés és én-tágítás, autonómia, ágencia, koherencia. E pszichés folyamatok ellentételezés nélküli túlsúlya a mentális zavarok sajátos formáihoz vezethetnek: például a mindent birtokolni akarás (értékek, bel- és külvilági tartalmak) túlzott formája az egocentrizmusban jelenhet meg, mi több, utalhat a nárcizmus és az autizmus veszélyes mértékű megnövekedésére is. A saját létezés kiterjesztésének dominanciája szintén veszélyt sejtet: megalomániában, vallási téveseszmékben és paranoiában egyaránt manifesztálódhat. A fenti megközelítés értelmében csupán a selfre vonatkozó nyelvi markerek finom arányait figyelembe véve distinkciót tehetünk a birtoklás és a létezés dominanciájával jellemezhető, illetve az énérzés rendkívüli csökkenésével, az én nagyfokú elszegényedésével leírható pszichés állapotok, s voltaképpen az ezeket megjelenítő narratívumok között. Ennek vizsgálatára született meg a LAS Verticum self-referencia modulja.

2.1 A self-referencia modul operacionalizálása

A self-referencia modul minden olyan kifejezés esetén találatot jelez, amelyben az első személy szerepel, legyen az ige (pl. nézek, látom), névmás (pl. én, enyém, magam), avagy birtokos személyjellel ellátott főnév (pl. könyvem). A self-referencia modul reliabilitására vonatkozó vizsgálat szövegkorpuszát a korábban már bemutatott minta alkotta. A kapott eredmények az alábbiak szerint alakultak: helyes találat 99%, téves riasztás 13%, míg a kihagyások értéke 8‰. A reliabilitásra vonatkozó adatok tükrében a self-referencia modul validitására vonatkozó vizsgálatok is elvégezhetők, amelyet a tagadás modulnál leírt metodika alapján hajtottuk végre. Hipotéziseink és az eredmények az alábbiakban olvashatók:

2.2 A self-referencia modul érvényességének vizsgálata

1) Egy számunkra fontos személlyel összefüggésbe hozható jó történet olyan élményfeldolgozást hoz felszínre, amelyben megjelenik az érzések, a gondolatok megoszthatósága, s ezzel egyidejűleg az „én"-nel szemben a „mi" válik hangsúlyossá, amelyben a kötődés, a kapcsolat érhető tetten. Azonban a veszteség-, a félelem- és a rossz történetek éppen ellentétes élményminőséget hívnak életre: az egyedüllétet, a társaktól való szeparációt, a kapcsolatok hiányát avagy frusztrált jellegét indukálják. A self-referencia vonatkozásában a teljesítménytörténetek szintén ez utóbbiakhoz hasonlóak, hiszen sikereink hátterében gyakran jelenik meg az „egyedül csinálás" öröme, amely autonómiával társul. Mindezek alapján különbséget várunk a self-referencia mennyiségi paraméterének vonatkozásában az élettörténeti narratívumok tematikája alapján. Az eredmények igazolják elvárásunkat: a teljesítmény-, a félelem-, a veszteség- és a jelentős személlyel kapcsolatos rossz történetekben a self-referencia relatív gyakorisága szignifikánsan nagyobb, mint a jelentős személlyel kapcsolatos jó történetekben ($M_{Selfref}$=0,1074 vs. 0,0799; t(71)=3,37, p<0,001).

2) A továbbiakban a LINTAG-gal, a self-referencia modul vonatkozásában a félelem-/szorongástörténetekben kapott gyakorisági értékek alsó és felső kvartiliseit kétmintás t-próbával hasonlítottuk össze a korábban már megismert személyiségvizsgáló eljárásokkal. A kapott eredmények alapján megállapítható, hogy a félelemmel, a szorongással együttjáró élethelyzetek olyan elbeszélési formát hívnak életre, amelyben a szituációval való megküzdés sikeres, avagy sikertelen volta a szöveg szintjén a self-referencia mennyiségi paramétereiben is kifejezésre jut. Az adott helyzet által kiváltott szorongással való megküzdés, azaz a düh, az elégedetlenség szabályozásának képessége (M_{BFQImp}=32,17 vs. 39,05; t(35)=-3,25, p<0,01) a self-referencia magasabb értékével jár együtt, ezáltal biztosítva a személy önfenntartását, a valósághoz való alkalmazkodását és a egyén integrációjának megtartottságát.

Bibliográfia

1. Freud, S. (1917/1997) Gyász és melankólia. In Ösztönök és ösztönsorsok. Metapszichológiai írások. Filum Kiadó, Budapest
2. Freud, S. (1925/1997) A tagadás. In Ösztönök és ösztönsorsok. Metapszichológiai írások. Filum Kiadó, Budapest
3. Kézdi, B. (1995) A negatív kód. Kultúra és öngyilkosság. Pannónia könyvek, Pro Pannonia Kiadó, Pécs
4. Murray, H. A. (1938) Exploration in personality. New York, Oxford Univesity Press
5. Szondi, L. (1956/1972) Lehrbuch der Experimentellen Triebdiagnostik. Band I. Verlag Hans Huber, Bern

A LAS VERTICUM 'Szereplő-funkció' modulja

Péley Bernadette

PTE BTK Pszichológiai Intézet,
7624 Pécs, Ifjúság útja 6.
peley@btk.pte.hu

Kivonat. Az előadás a LAS VERTICUM szó feletti narratív pszichológiai tartalomelemző szoftverrendszer 'Szereplők és funkciók' modulját mutatja be. A 'Szereplők' kódcsoport elemei: anya, apa, szülő, szűkcsalád, tágcsalád, nem rokonok. A 'Funkciók' kódcsoport összesen 21 elemből áll, mint például segítő, ellenség, elhagyó, antimodell, stb. A modul segítségével a vizsgált szövegekből automatikusan feltérképezhető a beszélő életdrámájában tematizált személyek és narratív funkcióik, valamint e tematizációk pszichológiai relevanciája, azaz más pszichológiai konstruktumokkal mutatott összefüggésrendszere.

A LAS VERTICUM szereplő-funkció modulja arra a feltételezésre épül, hogy a sajátos jelentésszervezési módoknak és maguknak a jelentéseknek diagnosztikus és prediktív értékük van az egyének és csoportok alkalmazkodó viselkedése szempontjából. Az elbeszélést (az elbeszélés képességét) a jelentésszervezés biológiailag adott, ám kulturálisan közvetített formájának tekinti, amelynek rendszeres tartalmi és formai mintázatai empirikus alapot nyújtanak az alkalmazkodó viselkedésre vonatkozó következtetések levonására. Ezzel összefüggésben a kutatás során fogalmilag és módszertanilag (a jelentős életeseményeket elbeszéltető narratív interjú technikáinak fejlesztése révén) szűkíteni és konkretizálni kívántuk a narratívum és a pszichológiai konstrukciók közötti, a narratív meta-elméletek által feltételezett összefüggéseket.

Saját élettörténetünk elbeszélése során megjelenítjük, hogy történeteink szereplőinek hozzánk (az énhez) milyen viszonya van. Ez a viszony, vagyis a partnereknek az én fenntartása és működése szempontjából betöltött funkciója (az, hogy az egyén miként éli át kapcsolatait és hogyan észlel másokat a vele való viszonyban) a korai tárgykapcsolatokra épül, és az én-szerveződés sajátos mintáit tükrözi. Az élettörténeti elbeszélésekben ezért a szereplőkhöz kötött funkciók típusait különítettük el arra a korábbi kutatási eredményünkre alapozva, hogy korai kapcsolati fejlődési zavarok a fenyegető, illetve szorongató típusú funkciókkal korrelálnak, míg a normális, kiegyensúlyozott én-fejlődésre a támogató-gondoskodó jellegű funkciók jellemzők (Péley, 2002). Az alábbi szereplők és funkciók nyelvi mintáit kívántuk a nyelvi elemző modulba foglalni:

Szereplők

- Kód: anya
- Kód: apa
- Kód: szülő
- Kód: szűkcsalád
- Kód: tágcsalád
- Kód. nem rokonok

Funkciók

Az alábbiakban néhány példával illusztráljuk a funkciókat és meghatározásaikat:

- Kód: *elhagyó*: a szereplő elhagyta az elbeszélőt válás, külön költözés, szakítás miatt.
- Kód: *ellenség*: a szereplő az elbeszélő szerint ellene fordul, összeveszik vele, vagy az elbeszélő számára az előzmények odavezetnek, hogy ő maga a szereplő ellen fordul.
- Kód: *felnőtt társ*: a szereplő az elbeszélőt annak felnőtt identitásában erősíti meg, ez konkrétabb az általános támogató funkciónál. Ez a funkció gyakorlatilag egy epizódra korlátozódik.
- Kód: *fenyegető*: a szereplő olyat mond, tesz, ami az elbeszélő lelki és/vagy fizikai létét fenyegeti.
- Kód: *korlátozó:* a szereplő az elbeszélőt fizikai vagy belső célok elérésében akadályozza.
- Kód: *modell:* a szereplő olyan tulajdonságokkal rendelkezik, illetve olyat tesz, ami az elbeszélő szempontjából példaként jelenik meg.
- Kód: *nemgondoskodó*: a szereplő az elbeszélő szerint nem vigyáz valakire, a biztonsághoz, a létezéshez szükséges feltételeket nem biztosítja.
- Kód: *segítő:* a szereplő a külső és/vagy belső célok elérésében az elbeszélő szerint előrevivő, aktívan résztvevő, gondoskodó.
- Kód: *szorongató:* a szereplő úgy jelenik meg az elbeszélő számára, mint szorongást keltő, ezt okozhatja konkrét tett és/vagy állapot.
- Kód: *támogató*: a szereplő az elbeszélő szerint valaki mellett aktívan jelen van, inkább egyengeti az útját, mint segíti.

További funkciótípusok: antimodell, áruló, drogostárs, elvesztett, nemtámogató, partner, sorstárs, társ, versenytárs, védelmező, védenc.

Nyelvi operacionalizálás

A nyelvi operacionalizálást a *fenyegető* funkció azonosításával szemléltetjük. Fenyegető funkcióban jelenik meg a szereplő az alábbi esetekben:

1. **Közvetlen fizikai, testi sértés igéi**: Kínoz, lő, rúg, szorít, szúr, tapos, üt, vág, ver, veret, zúz, csap, lapít, bánt, bántalmaz, büntet, támad, kivág, kizár, kihajít, stb.
2. **Igékhez kapcsolódó bizonyos előtagok esetén**: pl. agyon-, agyba-főbe-, stb.
3. **Mindenféle harcot, veszekedést kifejező ige, amely a szereplőtől az elbeszélőre irányul**: pl. kiabál, ordít, üvölt, veszekszik, piszkál, zaklat, ijesztget, fenyeget, bosszul, alávet, aláz, árt, áskálódik, stb.
4. **Közelítést kifejező igék** ('ad', 'elővesz', 'előkerül'), stb **+ ártó dolog** ('pofon', 'fegyver', 'kés'). Például: pofont adott, elővette a pisztolyt, előkerült a fakanál, stb.

A **szereplő-funkció** modul erősen kísérleti stádiumban van. A LAS VERTICUM szoftvercsomag további moduljainál azért nehezebb megvalósítani, mert a szereplők között finomabb felosztást követel (az elbeszélő, közvetlen rokon, távolabbi de szoros ismeretség...) és a viszonyok is összetettebbek (segítség, veszélyeztetés, távolodás...) A logika hasonló, de a finomabb szemantikai osztályozás nagy munkát igényel.

A kísérleti adatbázis elemei

A kísérleti adatbázis néhány mintaelem típusa a következő:

```
1 TARSZUK NPREL casedirection=TARGET casetype=BASE
neg=NO kod=SZUK

1 TARPARENT NPREL casedirection=TARGET neg=NO kod=ANYA

1 PARENT NPREL casetype=BASE caseaspect=POINT
caseprecisoin=PRECISE neg!=YES kod=ANYA

1 APR_NEARNID1 NA ownerpers=P1 casedirection=TARGET
casetype!=BASE kvantor!=NONE kvantor!=NO neg!=YES
```

Ezekből makrók segítségével adjuk meg a kívánt kategóriákat. Néhány példa a makrók formátumára:

```
DESERT = APR_FARVD3,APR_FARND1

SUPPORT = APR_FARVIX3,APR_NEARNI1

THREAT = APR_FARVD1,APR_FARND3
```

A szereplők funkciója modullal végzett vizsgálatok

Tekintettel arra, hogy a mindenképpen legbonyolultabb modul fejlesztése még kísérleti stádiumban van, reliabilitás vizsgálatokat nem volt értelme elvégezni. Az adott kutatási szakaszban meg kellett elégednünk olyan robosztus változók elemzésével, mint a pozitív (segítő, gondoskodó, támogató, stb.) és a negatív (elhanyagoló, korlátozó, fenyegető, stb.) funkciók összesítéséből képzett pozitív, illetve negatív szereplői funkciók. Ezeket a változókat viszonyítottuk azután a Beck-féle depresszió skálán mért eredményekhez, illetve a BFQ érzelemkontroll és impulzus kontroll skáláin mért eredményekhez. Az 1. táblázat mutatja, hogy a Depresszió skálán magas eredményt elért személyek történeteiben sok negatív szereplői funkció található, míg a pozitív funkciókkal megjelenő személyek száma alacsony.

1. táblázat. Szereplők funkcióinak összefüggése a depresszióval (n=83)

	Átlag	SD	Sign.
Depresszió Skála SUMNEG	4,024 1 6,180 7	3,5885 2 4,8743 8	,001
Depresszió Skála SUMPOZ	4,024 1 2,988 0	3,5885 2 2,9028 7	,044

A 2. táblázat az érzelmi kontroll és az impulzus kontroll skálák összefüggését mutatja azzal, hogy sok vagy kevés negatív szereplői funkciót említett-e a személy. A sok negatív funkció említése mindkét esetben gyengébb kontrollt jelez.

2. táblázat. Szereplők funkcióinak összefüggése az érzelmi kontrollal és az impulzus kontrollal (n=24)

	Átlag	SD	Sign.
ÉRZELMI KONTROLL	36,333 3	8,0036 2	,047
IMPULZUS KONTROLL	35,500 0	8,2935 3	,044

Még egyszer hangsúlyozni kell, hogy ezek az eredmények előzetes jellegűek, s csupán arra alkalmasak, hogy a további fejlesztés alatt álló modul pszichológiai tartalmát és nyelvi modul fejlesztésének elveit jelezzék.

Hivatkozás

Péley Bernadette (2002): *Rítus és történet. Beavatás és kábítószeres létezésmód.* Új Mandátum, Budapest

Narratív koherencia—elemző program helye a pszichológiai kutatásban

Papp Orsolya

PTE, Pszichológia Doktori Iskola, Pécs
1039, Bp., Juhász Gyula u. 58.
papporsi@lycos.com

Az előadásban egy "ideális számítógépes szoftver" tulajdonságait veszem nagyító alá egy narratív pszichológiai kérdésfeltevés szemszögéből, mely arra vonatkozik, hogy egy interjúhelyzetben elmondott személyes élettörténet mint szöveg számítógépes tartalomelemzése alapján milyen következtetéseket lehet levonni az elbeszélő mentális szerveződéséről. Elemzésem középpontjába az önéletrajzi szöveg strukturális tulajdonságai közül a koherenciát állítom; ezzel kapcsolatban első lépésben elkülönítem a nyelvi-, a szöveg- és a narratív koherencia fogalmát. Majd a szövegnyelvészeti kutatások eredményeire támaszkodva amellett érvelek, hogy a narratív koherencia megragadásához mindhárom koherenciatípus együttes kezelésére van szükség a számítógépes narratív pszichológiai tartalomelemzésekben.

A nyelvészetben és a pszichológiában is a hetvenes évek végétől került az érdeklődés középpontjába a koherencia fogalma (Kiefer, 1979; Kintsch, 1974 és 1977) – az előbbiben a szöveg folytonosságát megteremtő összefüggésrendszer kitüntetett strukturális jellemzőjeként, a pszicholingvisztikai kutatásokban pedig az emlékezeti felidézést segítő történetszerkezet és a mentális szerveződés mutatójaként (lsd erről részletesen, Pléh, 1986). A mai tudományos nyelvhasználatban ezeknek a hangsúlyoknak megfelelően szükségesnek látszik elkülöníteni a nyelvi-, a szöveg- és a narratív koherencia fogalmát (Komlósi, 2002). A nyelvi koherencia a nyelvhasználó grammatikai kompetenciájához kötődve az adott nyelv fonológiai, prozódiai, morfo-szintaktikai és logikai-szemantikai struktúráinak nyelvi környezetnek megfeleltetett megvalósítását jelenti. A szövegkoherenciát erre épülve az adott társas interakciók és cslekvési szabályszerűségek figyelembe vétele határozza meg. A narratív koherencia pedig ezekhez képest a beszélő mentális szerveződésének érzékeny mutatója: a világról alkotott fogalmi struktúráinak rendezettségét tükrözi, mely a mentális tartalmak gazdaságos működésének és megoszthatóságának kereteit biztosítja.

Az 1980-as évek közepétől egyre nagyobb teret nyerő narratív pszichológia a világ- és benne a megismerő én--reprezentációk szerveződésében kitüntetett szerepet szán a történetstruktúrának (Bruner, 1986; Gergen és Gergen, 1988; McAdams, 1988). Ebben a szemléleti keretben a kultúra által közvetített elbeszélésminták mentén szerveződő szelf vizsgálata elsősorban az élet jelentős eseményeiről szóló verbális beszámolók elemzésén keresztül valósul meg arra az előfeltevésre építve, hogy az élettörténet bizonyos jellemzői alapján következtetések vonhatóak le az elbeszélő személy aktuális szelfállapotára nézve. Az empirikus kutatások az önéletrajzi szövegek elemzésénél a

klasszikus narratológia kategóriáira támaszkodnak, melynek során a történet tér-idő szerkezetének (Erős, Ehmann, 1996), perspektívájának (Pólya, 2003), illetve a szereplők funkcióinak (Péley, 2002) az interjúszövegekben megjelenő mintázataihoz kísérelnek meg a szelfet jellemző pszichológiai minőségeket kötni. Magyarországon az MTA Pszichológiai Kutatóintézet Narratológiai csoportja (lásd László, Ehmann, Péley, Pólya, 2000) az élettörténeti szövegek elemzésében a strukturális mintázatok azonosításához elsősorban a szöveg explicit nyelvi jegyeinek automatikus, számítógépes tartalomelemző szoftver által végzett megragadására helyezi a hangsúlyt. Jelen dolgozat ebben a szemléleti keretben kíván támpontokat nyújtani a narratív koherencia problémájának árnyalásához, azt a kérdést állítva a középpontba, hogy milyen tulajdonságokkal kellene rendelkeznie egy narratív koherencia mérő szoftvernek.

1./Természetes nyelvi szövegek kezelése: mivel az empirikus kutatások elsősorban kétszemélyes önéletrajzi interjú-helyzetben felvett verbális élettörténeti beszámolókra épülnek, erőteljes elvárás az alkalmazott szövegelemző szoftverrel szemben, hogy elfogadható hatékonysággal legyen képes kezelni az élő nyelvre jellemző töltelékszavakat, elakadásokat, befejezetlen, esetenként agrammatikus mondatokat és a nagyszámú szóismétlést egyaránt.

2./ Szövegkoherencia vs. Narratív koherencia: a bevezetésben részletesen kifejtett különbségek ellenére láthattuk, hogy a narratív pszichológiai tartalomelemzés módszere éppen azon alapul, hogy explicit nyelvi elemek gyakorisági előfordulása és mintázata révén azonosítson az énreprezentáció minőségét tükröző átfogóbb szintű szövegszerkezeti tulajdonságokat. Egy narratív koherenciát mérő szoftvernek ennek megfelelően képesnek kell lennie megragadni az élettörténeti szövegek mikro- és makrostruktúráját egyaránt. Ehhez a magyar szövegnyelvészeti kutatások Petőfi S. János által képviselt iránya, mely a szövegkoherencia mutatóit a hagyományos nyelvi szinteknek (grammatika, szemantika, pragmatika) megfelelő felosztásban tárgyalja, hasznos kiindulópontként szolgálhat.

Nyelvi szintek	*A szövegösszefüggés kifejező eszközei*
Pragmatikai szint: Tárgy- és eseményleírásoknak egy elképzelt kontextusra (világrészletre) vonatkoztatott összefüggősége ***KOHERENCIA***	Kommunikációs partnerek, szereplők, nézőpontok; Szándék, cél; Háttérismeret, tudáskeret, forgatókönyv; Közös nyelvűség, közös nyelvváltozat ismerete; Nem-szándékos és szándékos félreértés.
Szemantikai szint: Tárgyak és események összefüggő jelentéslláncot alkotó megjelenítése. A szövegösszefüggést itt a téma változatlansága adja. Hatósugara szerint lehet **lineáris** (az egymás utáni szövegegységek lokális grammatikai-szemantikai összefüggése) és	**I. Globális kohézió**: *izotópia* (tematikus szövegháló), cím, fókuszmondat, tételmondat, keretmondatok, kulcsszó, szemantikai progresszió, kronotopológia (hely-időstruktúra)

Nyelvi szintek	*A szövegösszefüggés kifejező eszközei*
globális (a szöveg egészére kiterjedő strukturális, pragmatikai, stiláris szövegösszefüggés, amit a makroszerkezeti egységek jelentéskapcsolata hordoz). ***KOHÉZIÓ***	**II. Lineáris kohézió:** *Koreferens lexémák*: szóismétlés, szinoníma, hiperoníma-hiponíma, antoníma, helyettesítés, parafrázis, körülírás; Tematikus asszociációs mező; Mellérendelő kötőszók.
Szintaktikai szint: a szövegmondaton túlmutató olyan mondatgrammatikai eszközök, melyek segítségével a mondatokat szöveggé kapcsoljuk össze, illetve a szövegegész szintaktikai kötöttsége. ***KONNEXITÁS***	Névelőhasználat; *Koreferens kapcsolatteremtés*: névmásítás, egyeztetés; Ellipszis; Igeidő, igemód; Sorszámnév; Idézés.

A szöveg nyelvi szintjeit együttesen kezelő számítógépes elemző szoftver egyik központi modulja a koreferenciális viszonyok kezelésére irányulna, mely a táblázat szerint (a megfelelő részek dőlt betűvel szedve) magában foglalja a szintaktikai kötöttség pronominalizáció és egyeztetés révén megvalósuló részét és a téma egységességét biztosító fogalmi láncolat megragadását. Ez utóbbi egyben a globális szövegkohézió tematikai szerkezetét is kirajzolná. Ennek automatikus feltárására irodalmi szövegekben az elmúlt években több termékeny kísérlet történt (Petőfi, 1998; Boda, Porkoláb, 2002; Domonkosi, 2002; Kiss, 2002). Természetes nyelvi szövegek esetében az automatizált elemzések kiindulópontját a következő kategóriák képezhetnék: a kulcsszó feltárása, mely a szöveg tartalmi középpontjában álló, témát megjelölő szó, az adott interjúban köznyelvi átlagos előfordulásánál jóval magasabb szöveggyakorisággal jellemezhetően. Az ezzel kapcsolatban felmerülő probléma, hogy a gyakorisági összehasonlítás alapját milyen szövegkorpusz képezze: valamilyen általános adat a szó magyar nyelvi előfordulásával kapcsolatban vagy speciálisan önéletrajzi szövegek statisztikai elemzése. A kulcsszó kijelölése után következhetne a vele szemantikailag ekvivalens, koreferens fogalomszók megkeresése a szövegben, melynek legnyilvánvalóbb jele a változatlan vagy variált ismétlés (ez utóbbihoz szükséges egy automatikus morfológiai elemző használata). Az azonos jelentésosztályba tartozó, egységes fogalomkört kirajzoló szavak feltárása szinonímaszótárral lenne kivitelezhető, esetleg egy olyan modul létrehozásával, ami az elbeszélő egyén által a kulcsszóhoz asszociált szavakat tartalmazza.

Mindazonáltal a koreferencialitás elemzésén túlmenően a narratív koherencia megragadásánál speciálisan a történetre jellemző elemek feltárása is szükséges. Erre mutat rá a Memphisi Egyetem Pszichológiai Kutatócsoportja által kifejlesztett COH-METRIX szövegelemző program felépítése is, mely Kintsch szövegbefogadási modellje alapján hat különböző koherenciaelemző modult tartalmaz a kauzális, intencionális,

temporális, referenciális, téri és strukturális kohézió mérésére (Dufty, McNamara, Louwerse, Cai, Graesser, Internet).

Előadásomban arra a kérdésre kerestem a választ, hogy a szelffel identikusnak feltételezett élettörténeti szövegekben milyen automatikus szövegelemző program segítségével ragadható meg a narratív koherencia. A pszichológiai kérdésfeltevés által irányítottan két elvárás fogalmazódott meg egy ilyen számítógépes tartalomelemző programmal kapcsolatban: természetes nyelvi szövegek hatékony kezelése és a szövegkoherencia többszintű mérése. Ez utóbbi esetében a koreferencialitás szintaktikai és szemantikai összetevőinek, valamint a történetstruktúra speciális szerkezeti elemeinek azonosítási képessége nyújthatna biztos kiindulópontot a narratív pszichológiai kérdések megválaszolásában.

Felhasznált irodalom:

1. BODA, I. K., PORKOLÁB, J. (2002) Co-reference Analysis and the Structure of Natural Language Texts. In: Andor, J., Benkes, Zs., Bókay, A. (szerk.) Szöveg az egész világ. Tinta Könyvkiadó, Bp., 81-101.
2. BRUNER, J. (1986/2001) A gondolkodás két formája. Narratívák 5., 27-59.
3. DOMONKOSI, Á. (2002) A metaforikus szövegek koreferenciális elemzésének kérdései. In: Andor, J., Benkes, Zs., Bókay, A. (szerk.) Szöveg az egész világ. Tinta Könyvkiadó, Bp., 165-176.
4. DUFTY, D. F., MCNAMARA, D., LOUWERSE, M., CAI, Z., GRAESSER, A. C. Automatic Evaluation of Aspects of Document Quality. Internetes elérhetőség: csep.psyc.memphis.edu/cohmetrix
5. ERŐS, F., EHMANN, B. (1996) Az identitásfejlődés tükröződése az önéletrajzi elbeszélésben. In: Erős, F. (szerk.) Azonosság és különbözőség. Tanulmányok az identitásról és az előítéletről. Scientia Humana, Bp., 96-113.
6. GERGEN, K. J., GERGEN, M. M. (1988/2001) A narratívumok és az én mint viszonyrendszer. Narratívák, 5, 77–121.
7. KIEFER, F. (1979) Szövegelmélet-szöveggrammatika-szövegnyelvészet. Magyar Nyelvőr, 216-225.
8. KINTSCH, W. (1974) The representation of meaning in memory. Hillsdale, N. J. Erlbaum.
9. KINTSCH, W. (1977) On comprehending stories. In: Just, M. A., Carpenter, P. A. (eds.) Cognitive processes in comprehension. Hillsdale, N. J. Erlbaum, 33-62.
10. KISS, S. (2002) Szövegkoherencia és szövegértelmezés a lírában. In: Andor, J., Benkes, Zs., Bókay, A. (szerk.) Szöveg az egész világ. Tinta Könyvkiadó, Bp., 312-318.
11. KOMLÓSI, L. I. (2002) A nyelvi koherenciától a narratív koherenciáig. In: Andor, J., Benkes, Zs., Bókay, A. (szerk.) Szöveg az egész világ. Tinta Könyvkiadó, Bp., 340-348.
12. LÁSZLÓ, J., EHMANN, B., PÉLEY, B., PÓLYA, T. (2000) A narratív pszichológiai tartalomelemzés: elméleti alapvetés és első eredmények, Pszichológia, 20: 367-390.
13. MCADAMS, D. P. (1988/2001) A történet jelentése az irodalomban és az életben. Narratívák, 5, 157–175.
14. PÉLEY, B. (2002) Rítus és történet: Beavatás és a kábítószeres létezésmód. Új Mandátum, Bp.
15. PETŐFI, S. J. (1998) Koreferenciális elemek és koreferenciarelációk. Officina Textologica 2.,15-31.
16. PLÉH, Cs. (1986) A történetszerkezet és az emlékezeti sémák. Akadémiai Kiadó, Bp.
17. PLÉH, Cs. (1994/1998) A narratívumok mint a pszichológiai koherenciateremtés eszközei. In: Hagyomány és újítás a pszichológiában. Balassi Kiadó, Bp.

18. PÓLYA, T. (2003) A narratív szelf pszichológiai értelmezései. In: A narratív identitás kérdései a társadalomtudományokban. Osiris, Bp., 55-69.

Kapcsolati mozgások számítógépes nyelvészeti vizsgálata élettörténeti narratívumokban

Pohárnok Melinda

PTE BTK Pszichológiai Intézet, Pécs, Ifjúság útja 6.
pomelin@freemail.hu

Abstract. A narratív pszichológiai tartalomelemezés az élettörténeti narratívumok inherens jellegzetességének tekintheti a szereplők egymás viszonyában való mozgását. (Pohárnok, 2003) Feltételezzük, hogy e mozgás két alapvető irányultsága – közeledés ás távolodás – jól megragadható nyelvi struktúrákkal rendelkezik, amelyek a számítógépes nyelvi feldolgozás eszközeivel elérhetőek. Ugyanakkor ezen nyelvi elemek megfeleltethetőek a kapcsolati szabályozás és a szelf szabályozás pszichológiai változójának. Az előadás bemutatja a Morphologic Kft. által kifejlesztett LintagTi szoftver "közelítés-távolodás" moduljának működését, és sorra veszi azokat a validitás vizsgálatokat, amelyek a modul által vizsgált jelenség pszichológiai relevanciáját igazolják és a fejlesztés alatt álló modul sikerességét ígérik.

Bevezetés

Kutatásunk alapvetése, hogy létezik egy olyan interperszonális vagy interaktív tér, amely mindig az én és a másik viszonya alapján szerveződik: a tér két végpontját az én és a másik adja meg, és egymás viszonyában való mozgásuk a kapcsolat alapvető sajátosságának tekinthető. Ez a tér, illetve az elbeszélő és a szereplők ebben való mozgása kiemelkedik az elsősorban kapcsolati témájú élettörténeti narratívumokban, és így vizsgálhatóvá válik a gépi szövegfeldolgozás eszközeivel.

Az AproxChange modul felépítése

A munkánk során kifejlesztett AproxChange modul kétféle irányú mozgást – a közelítést (APROX) és a távolítást (DIVER) – vizsgálja, mind a narratívum valódi, mind pedig emocionális terében. Egyaránt számításba vesszük tehát a fizikai mozgást, és az érzelmi "mozgást"(pl. szeret VS gyűlöl).

A közelítés és távolítás nyelvi markerei olyan szintaktikai egységek, ahol meghatározott szemantikájú igék (pl. mozgatást jelentő ige – "kap") meghatározott szemantikai csoportokba (pl. jelentős mások – "apa") sorolható főnevekkel állnak együtt. Például: "**anyához** mindig **odabújtam**" – közelítés, " de **anyám** mindig **lökött magától**" – távolítás. Az igéket a modul négy csoportba sorolva, egyes és többes szám harmadik személyű alakjaikban ismeri fel. A névszók esetében az én jelentésű névszókat és ragozott alakjaikat – elsősorban névmásokat – vesszük számításba, másrészt a "jeletős

másik" jelentésű névszói kifejezések és a ragozott alakjaik alkotnak segédkategóriákat. A "jelentős másik" mindig az adott kapcsolati elbeszélésben emelkedik ki, így azokat a személyeket soroljuk ide, akik az elbeszélő számára elsődlegesek egy adott történetben. Előzetes vizsgálatok alapján – és a felhasznált korpusz függvényében – idekerülnek a családtagok (a nukleáris család tagjaitól kezdve a tágabb család tagjaig), a szerelmi partner és a barátok. A névmásítás következtében szembe kellett néznünk azzal, hogy nem mindegyik tagmondatban szerepeltek a keresett névszók, így a keresésbe bevonjuk az első és harmadik személyű ragozott névmásokat is. A találatok így veszítenek pontosságukból, de többet nyerünk, mint vesztünk.

A modul reliabilitásának vizsgálata

A modul érvényességét a kézi és a gépi kódolás összehasonlításával végeztük. Több különböző élettörténeti szöveget kódoltunk (össz szószámuk: kb.2000), és a kézi kódolásnál több független kódoló adatait használtuk fel.

Table 6. Az AproxChange modul eredményessége

APROX	Helyes találat	Kihagyás	Téves riasztás	**TOTAL**
Kézi Kódolás	46	-	-	**46**
AproxChange modul	28	-	31	**28 (60%)**
DIVER	Helyes találat	Kihagyás	Téves riasztás	**TOTAL**
Kézi Kódolás	27	-	-	**27**
AproxChange modul	12	15	5	**12 (44%)**

A modul validitásának vizsgálata

A modul működését elsőként klinikai és normál minta összehasonlításával kívántuk ellenőrizni. A vizsgálatban illesztett szocioökonómiai státuszú személyekkel, illesztett interjútechnikával készített és illesztett szószámú élettörténeti narratívumokkal dolgoztunk. Feltételeztük, hogy azonos élményminőséget hívó kapcsolati történetekben az egészséges és a borderline személyiségzavarban szenvedő vizsgálati személyek eltérő érzelmi szabályozási mintát mutatnak – tehát eltérő arányban jelenik meg náluk a közelítés-távolítás kifejezések megoszlása. Feltevésünk sze-

rint az instabilabb szabályozást a közelítés és távolítás formák együttes előfordulásának nagyobb száma jelzi a borederline személyiségzavarban szenvedőknél. Mahler (1975) és kurrens szerzők (pl. Holmes, 2004) elképzelésére alapozva a borderline betegeknél megfigyelhető fokozott ambivalencia saját belső állapotaik – érzelmek, indulatok – szabályozatlanságának felel meg.

Table 7.

JÓ TÖRTÉNET						
Vizsgált csoport	APROX		DIVER		APRDIV	
Kontroll (N=33)	M	SD	M	SD	M	SD
	3.12	2.58	**0.51**	0.97	**3.63**	2.80
Borderline (N=33)	**4.09**	3.52	**0.96**	1.31	**5.06**	4.02
ROSSZ TÖRTÉNET						
Vizsgált csoport	APROX		DIVER		APRDIV	
Kontroll (N=32)	M	SD	M	SD	M	SD
	4.28	3.20	**0.68**	0.96	**4.96**	3.47
Borderline (N=32)	4.09	3.40	**1.34**	2.05	**5.43**	4.64

A kiemelt változóknál az eredmények tendenciájukban ugyan megfelelnek elvárásainknak, de feltevésünket a vizsgálati és kontroll csoport szövegein végzett két-mintás t-próba nem erősítette meg. ($t=1.668$, $p>0.05$).

További hipotézisünk szerint az egészséges vizsgálati személyek esetében az Aprox és Diver kifejezések együttes előfordulásának növekedése jelezheti az affektív reguláció nehézségének mértékét. Ez intenzívebb "oda-vissza" mozgásban, a viselkedéses szabályozás – mentális szabályozással szembeni – túlsúlyában jelentkezik, és szoros összefüggést mutat a személyiséget vonásként jellemző kontroll- illetve szabályozó működésekkel.

Table 8. A Teljesítmény és a Veszteség történetben történetben kódolt közelítés-távolítás összérték (APRDIV) és a vizsgált személyiségjellemzők közti összefüggések

APRDIV		**Személyiségjellemzők**					
		Érzelmi kontroll		**Impulzus kontroll**		**TMMS: tisztaság**	
		M	SD	M	SD	M	SD
Teljesítmény APRDIV	**A**	39.43*	8.74	36.26*	6.07	47.10*	4.97
	M	29.71*	7.01	30.87*	7.57	40.66*	6.36
Veszteség APRDIV	**A**	38.03	8.03	36.86*	6.86	46.44	3.85
	M	33.27	10.55	32.63*	7.19	43.00	7.61

APRDIV		Személyiségjellemzők			
		PLS: kezelhetőség		BCS: bizalom	
		M	SD	M	SD
Teljesít-	A	14.56*	2.69	44.90*	7.06
mény	M	12.18*	2.94	39.50*	8.08
APRDIV					
Veszteség	A	14.24	2.66	45.75*	6.58
APRDIV		12.94	3.86	40.15*	8.88

* A megjelölt átlagok esetében a személyiségjellemző átlagai az alacsony illetve magas APRDIV értékeket mutató csoportban szignifikánsan különböznek.

Eredmények és összefoglalás

A 2. Táblázat eredményei szerint a gyengébb érzelmi kontrollal és gyengébb impulzuskontrollal bíró személyek több közelítés és távolítás kódot kapnak a Teljesítményről szóló élettörténeti elbeszélésben. Ugyanitt kevesebbszer fordul elő a közelítés és távolítás együtt azoknál a személyeknél, akik megérthetőnek, bejósolhatónak tartják az őket érő ingereket és adekvátan képesek kezelni őket. Mindez alátámasztja azon elképzelésünket, hogy a közelítés és távolítás kódok gyakoribb együttes előfordulása instabilabb és inadekvátabb szabályozási működést jelezhet. Mind a Veszteség-, mind a Teljesítmény témájú történetben magasabb partnerbe vetett bizalomról számolnak be azok, akiknél alacsonyabb a közelítés-távolítás együttes előfordulása: feltehetjük, hogy a megbízhatónak észlelt másikat az ilyen személyek sikeresebben "használják" saját állapotuk regulására, így nem kell kizárólag saját viselkedéses szabályozásukra hagyatkozniuk.

A modullal végzett vizsgálatok annak az ígéretnek a megvalósulását vetítik előre, hogy a számítógépes szövegelemzés segítségével kapott kvantitatív adatokat a pszichológiai jelentések világában alkalmazhatjuk.

Irodalom

Mahler, M.; Pine, F.; Bergman, A. (1975): *The Psychological Birth of the Human Infant.* New York, Basic Books.

Holmes, J. (2004): Disorganized attchment and borderline personality disorder: A clinical perspective. *Attachment and Human Development* vol.2. is.6 pp.181-190.

Pohárnok M. (2003): *A narratívum mint pszichodiagnosztikai és pszichoterápiás médium megközelítési lehetőségei.* (absztrakt) Pszichoanalízis és Narratívum (A PTE Pszichológiai Intézet Elméleti Pszichoanalízis PhD-program által szervezett műhelykonferencia) Pécs, pp.38

Élettörténeti narratív perspektíva és érzelemszabályozás

Pólya Tibor

MTA Pszichológiai Kutatóintézet
1132 Budapest, Victor Hugó u. 18-22.
polya@mtapi.hu

Kivonat. A cikkben meghatározom az élettörténeti narratív perspektíva fogalmát, formai variációit (visszatekintő, újraátélő és átélő formák), illetve azokat a nyelvi jegyeket, amelyek alapján azonosítható az élettörténeti narratívumban érvényesülő narratív perspektívaforma. Bemutatom az élettörténeti narratív perspektívaforma automatikus azonosítására a MorphoLogic Kft.-vel közösen kifejlesztett modult, és a modul megbízhatósági tesztjének eredményeit. Végül ismertetem az élettörténeti narratív perspektíva érzelem szabályozó funkciójának igazolására végzett vizsgálat főbb eredményeit.

Az élettörténeti narratív perspektíva meghatározása

Az élettörténeti narratívum elbeszélője – mint minden narratívum elbeszélője – rendszerint meghatározott nézőpontból mutatja be a narratívum tartalmát adó narratív elemeket (eseményeket, szereplőket és körülményeket). A nézőpont számos meghatározása közül Zubin és Hewitt (1995) deiktikus centrum elmélete alapján az élettörténeti narratívum nézőpontját az élettörténeti narratívum deiktikus centrumával azonosítom. Jelen megközelítés szerint az élettörténeti narratív perspektíva relációs fogalom, amely a deiktikus centrum és a narratív elemek közötti kapcsolatra vonatkozik. Ezt a kapcsolatot a deiktikus centrum és a narratív elemek idői elhelyezése alapján határozom meg, mivel a narratív struktúra a deiktikus centrum és a narratív elemek idői elhelyezését is magában foglalja. Azt feltételezve, hogy a deiktikus centrum és a narratív elemek idői elhelyezése két értéket vehet fel (múlt versus jelen) az élettörténeti narratív perspektíva három formája írható le (visszatekintő, újraátélő és átélő formák).

A deiktikus centrum és a narratív elemek idői elhelyezésének két alapvető módja különböztethető meg a narratív szövegben, az elbeszélőtől független illetve az elbeszélőtől függő mód. Az első esetben az elbeszélő személy a saját idői elhelyezésétől függetlenül adja meg a narratív elem idői elhelyezkedését. A második esetben az elbeszélő személy a saját idői elhelyezéséhez képest határozza meg a narratív elem idői elhelyezkedését. Az élettörténeti narratív perspektíva nyelvi meghatározásának alapja az, hogy szisztematikus megfelelés van az egyes formák és aközött, hogy a deiktikus centrum a narratív elemhez közeliként vagy távoliként lokalizált. Visszatekintő forma érvényesülése esetén a deiktikus centrum a narratív elemhez képest időben távoliként lokalizált, míg újraátélő és átélő formák érvényesülése esetén a deiktikus centrum a narratív elemhez képest időben közeliként lokalizált. Az élettörténeti narratív perspek-

tívaformák nyelvi meghatározása a nyelvi jegyek öt csoportjára terjed ki: idő, személy és hely deixis, az egyes narratív perspektívaformákhoz kapcsolódó kifejezések (dátumra utaló kifejezések, indulatszavak és a szubjektív modalitás kifejezései), és a mondat módja.

Az élettörténeti narratív perspektíva pszichológiai funkciója

A pszichológiai funkció korábbi vizsgálatai (Pólya, 2003; Pólya, László, és Forgas, 2004) azt mutatták, hogy az élettörténeti narratív perspektívának az elbeszélő személy aktuális érzelmi állapotának szabályozásában van szerepe. Visszatekintő narratív perspektívaforma érvényesítése esetén az elbeszélő személy aktuális érzelmi állapota koherens és alacsony intenzitású. Újraátélő és átélő narratív perspektívaformák érvényesítése esetén kevésbé koherens és magasabb intenzitású az elbeszélő személy aktuális érzelmi állapota. Ugyanakkor fontos különbség, hogy az átélő forma érvényesítése lehetőséget ad a narratív elemekhez kapcsolódó tapasztalatok minőségének megváltoztatására, így negatív események elbeszélésekor az elbeszélő személy aktuális érzelmi állapota pozitív minőségű. Az újraátélő forma érvényesítésekor azonban nincs lehetőség a minőség megváltoztatására, így negatív események elbeszélésekor az aktuális érzelmi állapot minősége is negatív.

Amennyiben az élettörténeti narratív perspektíva hatékonyan befolyásolja az elbeszélő személy aktuális érzelmi állapotának minőségét, feltehető, hogy a narratív perspektívaformák érvényesítése az érzelemszabályozás vonás jellegű jellemzőivel is kapcsolatban van.

Vizsgálatok

A narratív perspektíva modul reliabilitásának vizsgálata

A modul megbízhatóságát két módon vizsgáltam. 20 homoszexuális férfi (M= 30.80, SD=8.66) és 20 IVF kezelésben résztvevő nő (M= 34.25, SD=3.96) narratív interjújából 130 élettörténeti narratívumot választottam ki (15 700 tagmondat, 68 320szó), amelyekben az élettörténeti narratív perspektívaformát kézzel és a narratív perspektíva modullal is kódoltam. A narratív perspektívaformák kézi és gépi kódjainak abszolút gyakorisága közötti korreláció a visszatekintő forma esetében r_V=0.814, $p < 0.01$, újraátélő forma esetében $r_Ú$=0.602, $p < 0.01$, végül az átélő forma esetében $r_Á$=0.472, $p < 0.01$.

A szöveg minta 1 élettörténeti narratívumán (210 tagmondat, 790 szó) részletesebb elemzést is végeztem. A modul az érvényesülő narratív perspektíva formák 57.3 %-át kódolja helyesen. A visszatekintő forma azonosításában a legsikeresebb a modul (76.8%), ezt követi az újraátélő forma (49.1%), és az átélő forma azonosításában a leggyengébb (46.1%).

A reliabilitás vizsgálatok alapján azt állapíthatjuk meg, hogy a narratív perspektíva modul megbízhatósága közepes szintű.

Az élettörténeti narratív perspektíva érzelem szabályozó funkciójának validitása

A vizsgálatban 83 személy (29 férfi és 54 nő két életkori csoportból: 18-35 és 45-60 éves) vett részt. A résztvevők 6-6 élettörténeti narratívumot beszéltek el (első emlék, teljesítmény, veszteség, félelem, jó és rossz kapcsolati narratívum). Az érzelmi szabályozás vonás jellegű jellemzőinek vizsgálatára a Vonás Meta-Hangulat Skála (Salovey, et al., 1995) tisztaság faktorát, a Személyiség Jellemzők Kérdőív (Caprara, et al., 1993) érzelmi és impulzus kontroll faktorait, és a Tematikus Appercepciós Teszt történetei alapján kódolt érzelmi labilitást használtam. A személyek depresszióját a Beck Depresszió Kérdőívvel vizsgáltam. Az elemzésekben az élettörténeti narratív perspektívaformák relatív gyakoriságával számoltam, amelyet az adott forma előfordulásának és a három forma összes előfordulásának hányadosaként kaptam.

A nemek között nem várunk különbséget az egyes narratív perspektívaformák előfordulásában. Az elvárást igazolóan nincs különbség a férfi és női résztvevők között az egyes élettörténeti narratív perspektívaformák relatív gyakoriságában.
Az életkor alapján azonban várhatóan jelentkeznek különbségek. Mivel felnőtt személyek több eseményt idéznek fel az identitás kialakítása szempontjából fontos fiatal felnőtt kor időszakából, mint más élettörténeti szakaszokból, valószínű, hogy a fiatal résztvevők viszonylag friss eseményeket, az idősebb résztvevők viszont inkább távolabbi eseményeket idéznek fel. Az idősebbeknek feltehetően több alkalmuk volt az esemény elbeszélésére, és ez által az esemény jelentésének meghatározására. Ennek alapján az átélő forma ritkább előfordulását várhatjuk, mivel ez a forma a történet átalakításában vesz részt. Az idősebbek ugyanakkor egyre inkább az események érzelmi összetevőire figyelnek, ezért azt is várhatjuk, hogy gyakrabban érvényesítenek újraátélő formát. Az eredmények mindkét elvárást igazolják. Az idősebb személyek ritkábban érvényesítenek átélő narratív perspektíva formát ($t_{(81)}$=2.04, $p < 0.05$) és gyakrabban érvényesítenek újraátélő formát ($t_{(81)}$=2.54, $p < 0.05$), mint a fiatalabb személyek.

Az élettörténeti narratívum témája szerint szintén nem várunk különbséget az élettörténeti narratív perspektívaformák előfordulásában. Az élettörténeti narratívum témájától való függetlenséget megerősítően az első emlék, teljesítmény, veszteség, félelem, illetve jó és rossz kapcsolati narratívumokban nincs eltérés a narratív perspektívaformák relatív gyakoriságában.

A pozitív és a negatív eseményeket elbeszélő élettörténeti narratívumok között azonban lehet különbség. Pozitív esemény elbeszélője feltehetően növelni, negatív esemény elbeszélője pedig feltehetően csökkenteni kívánja az eseményhez kapcsolódó érzelmek intenzitását. Pozitív esemény elbeszélője ezért várhatóan több újraátélő és kevesebb visszatekintő és átélő narratív perspektívaformát érvényesít, mint a negatív esemény elbeszélője. A visszatekintő és újraátélő formák relatív gyakorisága megfelel az elvárt mintázatnak, de az eltérés egyik esetben sem szignifikáns.

Az érzelem szabályozás kérdőívekkel mért jellemzői és az élettörténeti narratív perspektívaformák előfordulása között számos összefüggés támogatja az élettörténeti

narratív perspektíva érzelem szabályozó funkcióját. A visszatekintő narratív perspektívaforma alacsony érzelmi intenzitását a depresszió és a jelentésteliség faktoron jelentkező összefüggések igazolják. Azok a személyek, akik alacsony pontszámot érnek el a depresszió ($t_{(81)}=1.87$, $p < 0.10$) és jelentésteliség ($t_{(81)}=1.77$, $p < 0.10$) faktoron, és így a rájuk jellemző érzelmi intenzitás is alacsonyabb, több visszatekintő narratív perspektívaformát érvényesítenek, mint akik magas pontszámot érnek el ezen a két faktoron. Ugyanakkor az érzelmi kontroll faktoron magas pontszámot elért személyek az elvárásokkal ellentétesen kevesebb visszatekintő formát érvényesítenek, mint az érzelmi kontroll faktoron kevesebb pontszámot elért személyek ($t_{(81)}=1.76$, $p < 0.10$).

Az újraátélő narratív perspektívaforma magas érzelmi intenzitását szintén a depresszió faktoron jelentkező különbség támasztja alá. Annak megfelelően, hogy a depresszióhoz magasabb érzelmi intenzitás kapcsolódik, a depresszió kérdőíven magas pontszámot elért személyek több újraátélő formát érvényesítenek, mint a kevésbé depressziós személyek ($t_{(81)}=2.62$, $p < 0.05$).

Az átélő narratív perspektívaforma átalakító szerepét is támogatják az eredmények. Az érzelmileg labilisabb személyek gyakrabban érvényesítenek átélő formát, mint az érzelmileg kiegyensúlyozottabb személyek ($t_{(81)}=1.84$, $p < 0.10$). Ugyanakkor a megérthetőség faktoron magas pontszámot elért személyek, akik az eseményeket koherens struktúrában észlelik több átélő formát érvényesítenek, mint a kevesebb pontszámot elért személyek ($t_{(81)}=2.08$, $p < 0.05$).

A vizsgálat eredményeit összefoglalva azt állapíthatjuk meg, hogy az élettörténeti narratív perspektíva és az érzelem szabályozás vonás jellegű jellemzői között talált összefüggések nagy része támogatja az élettörténeti narratív perspektíva érzelem szabályozó funkciójának validitását.

Hivatkozások

1. Antonovsky, A. (1987). Unraveling the mystery of health. How people manage stress and stay well. Jossey-Bass, San Francisco.
2. Caprara, G., Barbaranelli, C., Borgogni, L., & Perugini, M. (1993). The "Big Five Questionnaire": A new questionnaire to assess the Five Factor Model. *Personality and Individual Differences*, 15(3), 281-288.
3. Pólya, T., László, J., & Forgas, J.P. (2004) Making sense of life stories: The role of narrative perspective in communicating hidden information about social identity and personality, *European Journal of Social Psychology*. (megjelenés alatt)
4. Pólya T. (2003). *Az élettörténet narratív perspektívája és az elbeszélő személy identitás állapotának minősége*. Ph.D. disszertáció, PTE BTK, Pécs.
5. Salovey, P., Mayer, J.D., Goldman, S.L., Turvey, C., & Palfai, T.P. (1995). Emotional attention, clarity, and repair: Exploring emotional intelligence using the Trait Meta-Mood Scale. In J.W. Pennebaker *Emotion, disclosure, and health*. (pp. 125-154). American Psychological Association, Washington, D.C.
6. Zubin, D.A., & Hewitt, E.L. (1995). The deictic center: A theory of deixis in narrative. In J.F. Duchan, G.A. Bruder, & L.E. Hewitt (eds), *Deixis in narrative. A cognitive science perspective*. (pp. 129-155). Lawrence Erlbaum, Hillsdale.

VIII. Beszédfeldolgozás

Beszéd alapfrekvencia követés hatékony zöngésség detektálással

Bárdi Tamás

Pázmány Péter Katolikus Egyetem, Információs Technológia Kar
1083 Budapest, Práter u 50/A
bardi.tamas@itk.ppke.hu

Kivonat: A beszédjel alapfrekvenciát meghatározó algoritmusok, más néven pitch detektorok helyes működése csak úgy lehetséges, ha az automatikus zöngés-zöngétlen megkülönböztetés is megbízható. Az alábbiakban ismertetjük pitch detektorunkat, melyben a zöngésség detektálása a konkurens módszereknél alacsonyabb hiba százalékkal működik. Algoritmusunk a jól ismert autokorrelációs módszeren alapszik. Algoritmusunk zöngésség detektáló erejét egy olyan adatbázison vizsgáltuk, melyben a beszéddel szinkronban laryngográf jelet is rögzítettek.

1. Bevezetés

Az emberi hallás modern elméletei hitelt érdemlően megállapították, hogy a hangmagasság (pitch) észlelés nem mindig van egy-egy értelmű kapcsolatban az alapfrekvenciával (F0). Ennek ellenére a digitális beszéd-feldolgozásban az F0 becslő módszereket hagyományosan pitch detektor algoritmusoknak (PDA) nevezik. A tényleges beszéddallamot jól közelítő pitch kontúr sok alkalmazásban hasznosítható. Jelentős szerepe van a prozódikus elemzésekben. Ilyen például a mondat hangsúlyos helyeinek megtalálása a hanglejtés alapján, vagy a kérdő és kijelentő mondatok automatikus megkülönböztetése. A beszédfelismerés a tonális nyelveken, mint például a kínai vagy a vietnami, megoldhatatlan pitch detektor nélkül.

A szakirodalomban pitch detektor témában jó néhány módszer látott napvilágot az elmúlt évtizedekben [10], a legszélesebb körű áttekintésük Hess-nél olvasható [7]. A megoldások többsége mérsékelt teljesítményével elégedetlenségre adhat okot, de azért van néhány egészen jó is. Ilyen Bagshaw eSRPD [3, 4] módszere, amely kevesebb, mint 1%-ban becsli rosszul az alapfrekvenciát, ha zönge van a beszédben. De a zöngés gerjesztés meglétét vagy hiányát már 3-4% hibával detektálja.

Általánosságban elmondható, hogy nyelvtani jelentéssel bíró pitch csak a zöngés szegmentumokon figyelhető meg. Ezért pitch frekvencia meghatározásának feltétele a jó zöngésség detekció. A zöngés-zöngétlen megkülönböztetés (V/UV - voiced/unvoiced) szerepe a beszédfelismerésben is jelentős, hiszen számos olyan szópár van, pl. köt - köd, melyek kiejtésben csak egyik mássalhangzójuk zöngésségében különböznek.

Egy zöngésség meghatározására szolgáló algoritmus (VDA - voicing determination algorithm) gyakran implicit része egy PDA-nak vagy beszédfelismerőnek, de megvalósítható különállóan is. Számos VDA született [7] már különféle elméletek bevetésével, közülük néhány igazán figyelemre méltó, jó teljesítményt azonban csak nagyon kevés mutat. A pitch detektoroknál általában a V/UV tévesztések nagyobb százalékban fordulnak elő, mint az F0 becslési hibák. Atal és Rabiner [1, 2, 8] egy öt döntési paramétert használó VDA-val próbálkozott statisztikus mintázat-felismerési megközelítést alkalmazva. Módszerük 4%-os hibaarányt adott egy nehezebb feladat megoldásában, nevezetesen a zöngés/zöngétlen/csendes (nincs beszéd) (V/U/S - voiced/unvoiced/silent) osztályozásban az egyszerűbb zöngés/zöngétlen (V/UV) döntés helyett.

Egy PDA-t építettünk, melyben egy hatékony beépített zöngésség detektor működik. Algoritmusunk az autokorreláció függvényen (ACF) alapszik. A zönge detekcióban módszerünk 2%-hoz közeli hibaarányt ért el. Az algoritmus, ha az ACF számításához FFT-t alkalmazunk, kevesebb, mint 2 megaflop per szekundum processzszorigénnyel megvalósítható 8 kHz-es mintavételezés mellett.

Az alábbi szakaszok az algoritmus moduláris szerkezetének megfelelően szerveződtek. A 2. szakasz az előfeldolgozó részt tárgyalja. Preprocesszorunkat úgy terveztük, hogy a V/UV megkülönböztetést a lehető legjobban segítse, az említett hibaarány elérésében nélkülözhetetlen szerepet játszik.

Az előfeldolgozás után a beszédjelből rövid időtartamú szakaszok kerülnek a basic extractor-nak nevezett egységhez. Itt számítjuk az ACF-et, majd ebből nyerjük a V/UV döntéshez és az F0 becsléshez szükséges paramétereket. Ebből a részből "halszálka" módszer alkalmazása érdemel említést, amely az "F0 a felső limiten" típusú hibákat csökkenti. Mindezeket a 3. szakasz tárgyalja.

Az egyszerű, de hatékony beépített VDA részletezése és kiértékelése a 4. szakasz és egyben cikkünk fő tárgya. A V/UV döntés két paraméteren alapszik, mindkettőt egy-egy küszöbbel hasonlítjuk össze. Ez a kétküszöbös módszer szintén hozzájárult a hibaszázalék csökkenéséhez. A szakirodalomban szokásos az előállított pitch kontúrok utólagos simítására egy posztprocesszort alkalmazni. Ilyet mi nem használtunk, mert a vizsgálatunk célja a beépített VDA képességének felmérése volt. A kiértékelésben a fókuszt a megbízható zöngésség detektálásra helyeztük.

2. A beszédjel előfeldolgozó

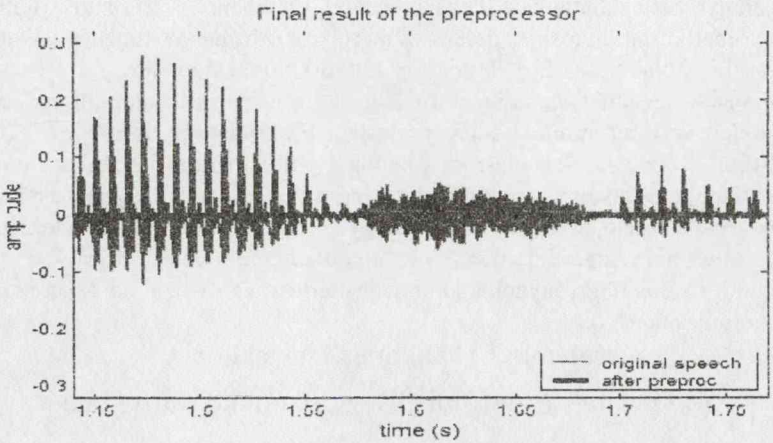
Általában egy PDA három fő komponensből épül fel: 1) preprocesszor, 2) basic extractor, 3) posztprocesszor. A preprocesszor fő feladata úgy transzformálni a beszédjelet, hogy utána az F0 becslés és a zönge detektálás könnyebb legyen.

A basic extractor rendszerint a beszédjelből vett tipikusan 20-50 milliszekundumos ablakokon dolgozik. A megkülönböztetés azonban, hogy mely műveletek tartoznak a preprocesszorhoz és melyek a basic extractor-hoz nagyon gyakran csak formális jelentőségű. Ha előbb kivesszük az ablakot a beszédjelből, majd azon futtatjuk a preprocesszort, akkor egyrészt fölöslegesen duplikálunk egy csomó számítást, ha az ablakok átfedik egymást, másrészt a preprocesszor és a basic extractor munkáját nehéz lesz külön-külön vizsgálni. Ha így teszünk, nem tudjuk például összefüggően meghallgatni a preprocesszorból kijövő jelet. A javaslatunk, hogy inkább futtassuk a

preprocesszort a beszédjel teljes hosszában, majd ebből vegyünk ablakokat és küldjük őket a basic extractor-hoz elemzésre. Ha így teszünk, érzékszervileg megfigyelhetővé válik a rendszer egy belső állapotában. Érzékszervi ellenőrző pontok elhelyezése egy összetett beszédfeldolgozó rendszer belsejében segítheti az empirikusan optimálandó paraméterek szerencsés megválasztását.

Preprocesszorunkban alul-áteresztő szűrést és ún. centerclip-et, magyarul középre vágást használunk. Mindkettő igen elterjedt a pitch detektorok szakirodalmában [6, 9, 11]. Az alul-átengedő szűrőnk Csebisev I-es típus, a levágási frekvencia 1200 Hz.

Az adaptív középre vágás technikája időben változó vágási szintet alkalmaz, mely a jel amplitúdójának függvényében változik. Általában ez a változó középre vágási szint a beszédjel valamilyen burkolójának egy rögzített százaléka. A módszerünkben az újítás, hogy kombinálja a két lépést, az alul áteresztő szűrést és a középre vágást. A burkolót az eredeti beszédjel amplitúdójából számítjuk, majd ennek 40%-át alkalmazzuk változó középre vágási szintként, de már a szűrt jelen. Mivel a tisztán sztohasztikus gerjesztésű beszéd szegmentumokon általában ennél nagyobb a nagy frekvenciás komponensek részaránya, a módszerünk a zöngétlen mássalhangzókat gyakorlatilag mindenütt nullára redukálja (1.ábra). Ez az effektus jelentősen javítja az automatikus V/UV döntés esélyeit.



1.ábra: Az eredeti beszédjel és a preprocesszor kimenete.

3. A basic extractor

A PDA-nak ez a része először a beszéd ablak autokorreláció függvényét számítja ki, majd az algoritmus az ACF "legjobb" csúcsát keresi meg. Az ACF értéke a kiválasztott csúcsnál, mint a periodicitás egy mértéke a zöngésség detektálására szolgál, a csúcs eltolási ideje pedig a periódus időt becsli. De hogy találjuk meg a "legjobb" csúcsot? Amint azt a későbbiekben látni fogjuk, a "legjobb" lokális maximum koránt sem feltétlenül globális is egyben.

Elöljáróban megjegyezzük, hogy az összes itt leírt képletben az idő dimenziójú változók és konstansok (τ, t, u, W) másodpercben értendők, a beszédjel kezelése analóg: integrálokkal, folytonos idővel és amplitúdóval. Az amplitúdót a rendszerben feldol-

gozható maximális amplitúdó arányában jelöljük: $-1.0 \le x(t) \le 1.0$. A fenti jelölésekkel biztosítjuk a tárgyalás függetlenségét a mintavételi frekvenciától és bitmélységtől. Konkrét alkalmazásban a mintavételi frekvencia és a minták számábrázolása ismeretében formuláink könnyen a megfelelő digitális változatra konvertálhatók.

A rövid távú autokorrelációnak a jelfeldolgozásban gyakran használt "rézsútos" (biased) definíciója helyett de Cheveigné [5] javaslata alapján annak "egyenes" (unbiased) definícióját használjuk, majd az ACF-et mesterségesen lejtősítjük. (W az ablak hossza, a vizsgálat során 32 ms-t használtunk)

$$r_t(\tau) = \frac{\int_{t-W/2}^{t+W/2} x(u)x(u-\tau)du}{\int_{t-W/2}^{t+W/2} x(u)^2 du} \quad (\tau, t, u, W \text{ szekundumban}) \quad (1)$$

és a mesterséges lejtés (a *gr* tényezővel szabályozhatjuk az erősségét):

$$r_t^{biased}(\tau) = r_t(\tau) \cdot (1 - gr \cdot \tau) \quad (2)$$

Az ACF lejtése oktáv tévesztés elkerülése miatt fontos, így a tényleges alapperiódusnak előnyt biztosíthatunk a többszöröseivel szemben. A "rézsútos" definíció a lejtést automatikusan biztosítja, de ennek mértéke kizárólag W-től függ. A mesterséges lejtéssel az ablak hossz és a "lejtőszög" külön-külön hangolható.

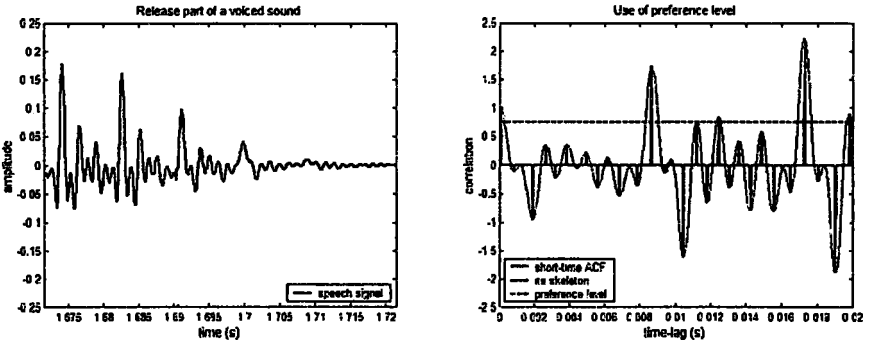
Mélyhangok kezdeti szakaszán az ACF gyakran a keresési intervallum szélén nagyobb értéket vesz fel, mint az alapperiódusnál. Ez a jelenség okozza az "F0 a felső limiten" típusú hibákat. Megoldási javaslatunk a problémára a "halszálka" módszer, a szkeleton függvény alkalmazása. Egy függvény szkeletonja a függvény értékét veszi fel annak lokális szélső értékeinél és nullát egyébként. Itt a céljainknak a lokális szélső érték szigorú és nem szigorú definíciói közötti átmenet felel meg.

Definíció: *f* valós függvénynek lokális szélső értéke van *x*-ben, ha *x*-ben nem szigorúan monoton és nem sík.

Definíció: $g = skeleton(f)$ akkor és csak akkor

$$g(x) = \begin{cases} f(x) & \text{ha } f\text{-nek lokális szélső értéke van } x\text{-ben} \\ 0 & \text{egyébként} \end{cases} \quad (3)$$

A mesterséges lejtés ellenére a tisztán zöngés hangok elhalkuló végein az ACF hajlamos a tényleges alapperiódus idő többszöröseinél egyre növekvő csúcsokat mutatni, amint az a 2. ábrán látható.



2. ábra: Egy magánhangzó elhalkuló vége és annak autokorrelációja.

Ez a jelenség csak olyankor fordulhat elő, ha az ACF a periódus időnél 1-hez közeli vagy afölötti értéket vesz fel. Ezért a probléma megoldására egy preferencia szint bevezetését javasoljuk. Az algoritmus válassza az első csúcsot, ami a preferencia szintet meghaladja. Ha ilyen nincs, akkor a legmagasabb csúcsot. Mi tapasztalati alapon 0.75-öt használtunk preferencia szintként.

Összegezve a basic extractor algoritmusunk lépései a korrekt végrehajtási sorrendben a következők:

Step 1: Az ACF kiszámítása (2) szerint.

Step 2: Szálkásítás:

$$sr_t(\tau) = skeleton(r_t(\tau))$$

Step 3: A keresési tartomány korlátozása (limited skeleton):

$$\text{Legyen } [F0_{\min}; F0_{\max}] \text{ a keresési intervallum,}$$

$$srl_t(\tau) = \begin{cases} -0.5 & \text{ha } \tau < 1/F0_{\max} \\ sr_t(\tau) & \text{ha } 1/F0_{\max} \le \tau \le 1/F0_{\min} \\ -0.5 & \text{ha } \tau > 1/F0_{\min} \end{cases} \tag{4}$$

Step 4: Mesterséges lejtés:

$$srl_t^{biased}(\tau) = (1 - gr \cdot \tau) \cdot srl_t(\tau); \quad \text{ahol gr=1.75} \tag{5}$$

Step 5: F0 becslés.

Step 5/A: Preferencia szint alkalmazása:

$$\tau^* = \min\{\tau : srl_t^{biased}(\tau) \ge 0.75\} \tag{6}$$

Step 5/B: Ha 5/A sikertelen, válasszuk a legmagasabb csúcsot:

$$\tau^* = \arg\max_{\tau}\{srl_t^{biased}(\tau)\} \tag{7}$$

és ekkor az alapfrekvencia:

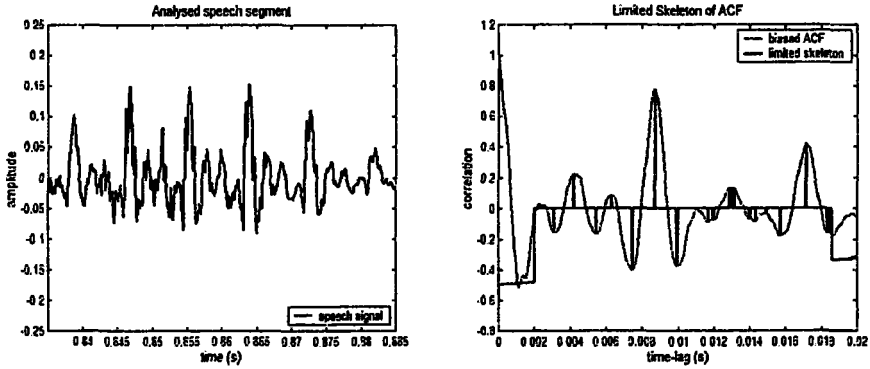
$$F0^* = \frac{1}{\tau^*}. \tag{8}$$

Step 6: A V/UV döntési paraméter:

$$rm_t = srl_t(\tau^*) \tag{9}$$

az "egyenes" (unbiased) korlátozott (limited) szkeletonból.

A 3. ábra mutatja az algoritmus működését.



3. ábra: Az *srl* (limited skeleton) maximuma mutatja a beszéd ablak alapperiódusát.

4. Zöngés-zöngétlen megkülönböztetés

Zöngésség detektorunk rm_t paramétert (9) használja döntése meghozatalában, valamint a jel energia logaritmusát:

$$p_t = 10 \cdot \log_{10}\left(\frac{1}{W}\int_{t-W/2}^{t+W/2} x(u)^2\,du\right) \quad \text{(decibel)} \tag{10}$$

A definícióból következik, hogy a maximális amplitúdójú négyszögjelre $p_t = 0\,\text{dB}$.

Ezek után a VDA egyszerűen összehasonlítja a paramétereket egy-egy küszöbbel. A zöngésség indikátor függvény pedig:

$$voicing(t) = \begin{cases} 1 & \text{ha } (rm_t > rmth)\,\&\,(p_t > pth) \\ 0 & \text{minden más esetben} \end{cases} \tag{11}$$

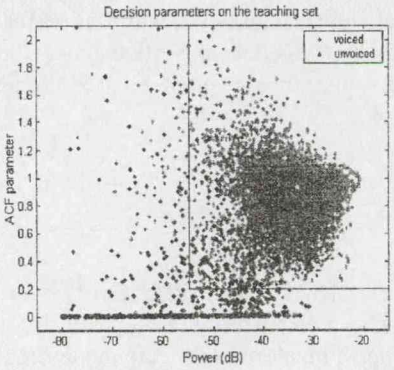
Ahol *rmth* és *pth* a küszöbök.

A kulcskérdés a továbbiakban a küszöbök optimális megválasztása. A hangolási folyamatot egybe kötöttük a döntési hibaarány kiértékelésével. A kiértékelésre szolgáló adatbázist két részre osztottuk: az egyik felén a betanítást, a másik felén az ellenőrzést végezzük. Tanításkor a küszöböket optimáljuk az adatbázis első felén, a másik felén pedig ellenőrizzük a VDA-t az optimált küszöbökkel. Természetesen az adatbázis két fele nem tartalmazhat közös részt, ez meghamisítaná a kiértékelést. A tanító és a teszt halmazba vegyesen tettük a női és férfi beszéd felvételeket, hogy az optimalizáció lehető legnagyobb beszélőfüggetlenséget biztosítsa.

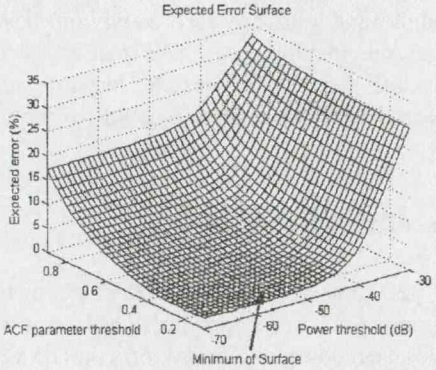
A döntési paraméterek kinyerése a teszt során W=32 ms ablakhosszal történt. Az $F0$ keresési tartomány 55 és 480 Hz között volt. A 4/a. ábra mutatja a paraméterek eloszlását a tanító halmazon. A világos pontok jelölik a zöngés, a sötétek a zöngétlen

szakaszokból származó paraméter párokat. A köztük haladó egyenes vonalak a kétküszöbös döntési módszert (11) ábrázolják. A vonalakon túlra tévedt sötét és világos pöttyök mutatják, hogy ez a módszer sem tökéletes.

A kétváltozós várható hibaarány felület az eloszlásokból származik. A felület értéke az *(x,y)* pontban azt jelenti, hogy *rmth=x* és *pth=y* küszöböket választva ennyi a V/UV tévesztés aránya a tanító halmazon. A felület mélypontja jelöli az optimális küszöböket. A 4/b ábrán látható a várható hibaarány felület.



4/a ábra: A döntési paraméterek eloszlása.



4/b ábra: Várható hibaarány felület.

Az optimált küszöbök: $pth = -55.2dB$ és $rmth = 0.23$. A hibafelület értéke ebben a pontban 1.95%. A kapott küszöböket teszteltük az adatbázis másik felén és a V/UV tévesztési arány **2.13%.** Ezt a hibaszázalékot, mint végeredményt tekinthetjük, ez az algoritmusunk teljesítménye.

5. Összegzés

Áttekintve az algoritmusunkat úgy látjuk, három jó részmegoldás játszott kulcsszerepet a 2.13%-os hibaarány elérésében. Az első az alul-áteresztő szűrés kombinálása a center clip-pel, a másik szkeleton függvény használata a basic extractor-ban, a harmadik pedig a jel energia figyelembe vétele a zöngésség meghatározásban. A jel energia sokkal jobban jelzi a zöngét, ha azt az előfeldolgozó után mérjük, mint ha az eredeti beszéden. Az algoritmus precíz megfogalmazása és a korrekt végrehajtási sorrend szintén lényeges.

Algoritmusunk implementálható valós idejű alkalmazásban is. Ekkor az algoritmusból fakadó (nem kiküszöbölhető) késés elsősorban az ablakszélességtől függ. 32 ms-os ablakot használtunk, ez 16 ms késést okoz. Ehhez még a burkoló számítás és az alul áteresztő szűrés legfeljebb 5 ms-ot tesz hozzá. A közeli jövőben elkészítünk egy PC-n futó valós idejű demo alkalmazást.

6. A kiértékelés adatbázisa

Algoritmusunkat a Fundamental Frequency Determination Algorithm Evaluation Database (FDA) elnevezésű beszéd adatbázison ellenőriztük. Ezt a University of Edinburgh egyetem Centre for Speech Technology Research intézetében készítették. A szerzője Paul Christopher Bagshaw. Az adatbázis letölthető az Internetről, az URL: http://www.cstr.ed.ac.uk/~pcb/fda-eval.tar.gz. 7 percnyi beszédet tartalmaz. 50 angol mondat, mindegyik egy férfi és egy női beszélő elmondásában. A tejes idő 37%-ában zöngés szegmentumok és 63%-ban zönge nélküliek (zöngétlen mássalhangzó és beszédszünet együtt). A beszédet laryngográf jellel szinkronban vették fel. Ez alapján címkézték a zöngés és zönge nélküli szegmentumokat.

Köszönetnyilvánítás

A szerző szeretné köszönetét kifejezni témavezetőjének, Dr. Takács Györgynek a iránymutatásáért és segítségéért, valamint a Pázmány Péter Katolikus Egyetem Információs Technológiai Kar doktori iskolája vezetőinek a bizalomért és a támogatásért.

Bibliográfia

1. B. S. Atal and L. R. Rabiner "A Pattern Recognition Approach to Voiced—Unvoiced—Silence Classification with Applications to Speech Recognition" IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP-24, pp. 201—212 (1976)
2. B. S. Atal and L. R. Rabiner: "Voiced-unvoice decision without pitch detection" J Acoust. Soc. Am., Vol. 58 (1975)
3. P. C. Bagshaw Automatic prosodic analysis for computer aided pronunciation teaching PhD Thesis, Univ. Edinburgh (1994)
4. P. C. Bagshaw, S. M. Hiller and M. A. Jack "Enhanced pitch tracking and the processing of F0 contours for computer aided intonation teaching" Proc. 3rd European Conf. on Speech Comm. and Technology, Vol. 2, pp. 1003—1006, Berlin (1993)
5. A. de Cheveigné and H. Kawahara: "YIN, a fundamental frequency estimator for speech and music" J Acoust. Soc. Am., Vol. 111, Apr (2002)
6. J. R. Deller, J. H. L. Hansen and J. G. Proakis Discrete-Time Processing of Speech Signals, Macmillan, New York (1993)
7. W. A. Hess Pitch Determination of Speech Signals, Berlin, Springer-Verlag (1983)
8. L. R. Rabiner "Evaluation of a statistical approach to voiced-unvoiced-silence analysis for telephone quality speech" Bell Syst. Tech. J., Vol. 56, pp. 455—482 (1977)
9. L. R. Rabiner "On the Use of Autocorrelation Analysis for Pitch Detection" IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP-25, pp. 24—33 (1977)
10. L. R. Rabiner, M. J. Cheng, A. E. Rosenberg and C. A. McGonegal "A Comparative Performance Study of Several Pitch Detection Algorithms" IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP-24, pp. 399—418 (1976)
11. L. R. Rabiner and R. W. Schafer Digital Processing of Speech Signals, Prentice Hall, Engelwood Cliffs NJ (1978)

Audiovizuális beszédfelismerés

Czap László

Miskolci Egyetem, Villamosmérnöki Intézet, Automatizálási Tanszék
3515 Miskolc, Egyetemváros
czap@mazsola.iit.uni-miskolc.hu

Abstract. Az emberi beszédértés bimodális természetű: az akusztikus és vizuális jelet zseniálisan kombináljuk a maximális érthetőség érdekében. Különösen zajos környezetben segíti a beszéd jobb megértését a vizuális jel. A szájról olvasás feladatát próbálom gépi úton megvalósítani. Az audiovizuális beszédfelismerés fő kérdései, hogy mely jellemzők hordozzák a lényegi vizuális információt, és hogy ezek hogyan nyerhetők ki a képből. A geometriai és pixel bázisú lényegkiemelést a folyamatos beszédfelismerés szempontjai szerint még nem hasonlították össze. Arra a kérdésre is választ kerestem, hogy eséllyel léphet-e fel a diádok vetélytársaként a félszótag, mint a felismerés alapegysége.

1. Bevezetés

Az emberi kommunikáció multimodális természetű. A multimodalitást értelmezhetjük úgy, hogy a kommunikációban több érzékszervünk vesz részt. A hallás mellett a látás a legfontosabb információforrásunk. Bernsen [1] összekapcsolja a modalitást a médiummal, mint az információ valamely formájának fizikai hordozójával. A média rokonítható az érzékszervekkel, amelyekre hat. A grafikus médium pl.: a látással, az akusztikus médium a hallással társítható. A technika fejlődésével az ember-gép kommunikációban is egyre több modalitás juthat szerephez.

Cikkem arról a munkáról számol be, amelynek keretében a vizuális modalitás által hordozott információval egészítem ki az akusztikus modalitás elemzését, a szájról olvasást próbálom gépi úton megvalósítani. Massaro [3] kísérletekkel igazolta, hogy a modalitásokat egymás kiegészítésére használjuk. Ha a hang gyenge minőségű, vagy hallássérült a megfigyelő, jobban hagyatkozik a szájról olvasásra. „Jobban hallom a TV-t, ha felteszem a szemüvegem." Az emberi beszédértést meg sem közelítő gépi felismerőket hasonlíthatjuk a környezet vagy képességei által korlátozott emberi felfogóhoz abban a tekintetben, hogy a kiegészítő vizuális jel a gépi beszédfelismerők felismerési hatékonyságát is javíthatja, különösen zajos környezetben.

2. Az emberi bimodális beszédfelismerés analízise

Napjainkban a gépi beszédfelismerés szédületes ütemű fejlődésének vagyunk tanúi. Ezek megbízhatósága azonban jelentősen romlik zajos beszéd esetén. Egyik kitörési pont lehet a vizuális jel felhasználása a beszéd felismeréséhez. A gépi szájról olvasás tervezéséhez célszerű ismerni, hogy az emberi kommunikációban az arc mely részei mennyire segítik a beszéd felismerését. Ezeket a vizsgálatokat zajos beszéddel végezhetjük el, hiszen a jó minőségű beszéd tökéletesen érthető, a vizuális támogatás hatása nem mérhető.

Ebben a fejezetben arra keressük a választ, hogy az arc mely részei hordozzák a legtöbb vizuális információt a beszédfelismeréshez. Érthetőség vizsgálatot végeztünk zajos beszéddel úgy, hogy az arc egyes részeinek maszkolásával a beszélő arcának csak részletei voltak láthatók.

Várakozásaink szerint az ajakforma lényeges vizuális jellemző. Fontos kérdés, hogy a nyelv és a fogak láthatósága számottevő javulást okoz-e, érdemes-e a lényegkiemelésnél erőfeszítéseket tenni a leírásukra. Az arc egyéb részei is hozzájárulnak a beszédfelismerési eredmények további javulásához, ha a beszélő egész arca látható? Ezekre a kérdésekre kerestük a választ szubjektív teszt segítségével.

2. 1. Érthetőség vizsgálat

Az arc különböző részletei által hordozott vizuális támogatás mérésére érthetőség vizsgálatot végeztünk. Mássalhangzó felismeréshez V_1CV_1 szavak (pl.: eke, ata) középső mássalhangzóját kellett megjelölni. Magánhangzó felismerésekor C_1VC_1 szavak (pl.: lol, tet) középső magánhangzóját kellett kitalálni.

A teszteket 78, fonetikai ismeretekkel nem rendelkező egyetemi hallgató töltötte ki. A 10-15 fős csoport egy közös TV készüléken nézte a videó jelet és egy közös hangszóróból hallotta a hangot, kétszer egymás után. A válaszra korlátozott idő – kb. 3 másodperc – állt rendelkezésre.

A vizuális jel az alábbiak valamelyike lehetett:

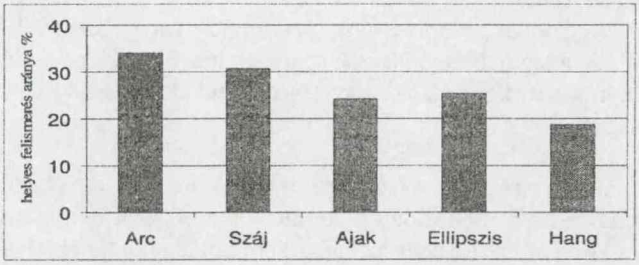
- a beszélő arca
- a beszélő szája (ajkak, fogak, nyelv)
- a beszélő ajkai
- az ajkak méreteit utánzó ellipszis

Az akusztikus jel zajos beszéd volt. A magánhangzó felismerési kísérleteket -18 dB pillanatnyi jel-zaj viszony mellett végeztük. A mássalhangzó felismerési vizsgálatoknál a jel-zaj viszony -6 dB volt. A pillanatnyi jel-zaj viszonyhoz szükséges zaj amplitúdót 5 ms-onként állítottuk be.

Egyes kísérleteknél csak akusztikus jel volt jelen (sötét képernyő), máskor hang nélkül, csak a kép alapján próbáltuk a magánhangzót vagy a mássalhangzót felismerni.

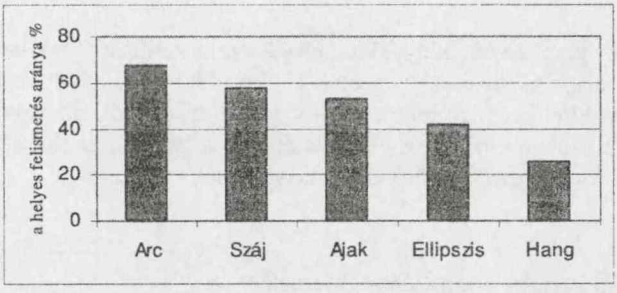
Az eredmények 11 623 válasz kiértékelése alapján születtek, ezek közül 9 625 a mássalhangzó, 1 998 a magánhangzó felismerését szolgálta. Egy hallgató egy hang megjelölésével adott egy választ.

Az 1. ábra a mássalhangzó érthetőség vizsgálat eredményét mutatja audiovizuális jel esetén.



1. ábra. Mássalhangzók felismerési aránya. A hang pillanatnyi jel-zaj viszonya: -6 dB, a képen az arc részletei, vagy a száj vízszintes és függőleges méretével megegyező kis- és nagy tengelyű ellipszis volt látható. Az Hang megnevezésű oszlop esetében a képen csak a minta sorszáma volt látható.

A 2. ábrán a magánhangzók felismerési eredményeit láthatjuk.



2. ábra. A magánhangzók felismerési arányai. A pillanatnyi jel-zaj viszony -18 dB.

Várakozásunkkal egyezően, minél többet látunk a beszélő arcából, annál jobban segíti a kép a beszéd felismerését. Mivel jelentéssel nem bíró szavakról van szó, az arckifejezés nem növelhette az érthetőséget az egész arc megmutatása esetén sem. A javulás a száj (ajkak, fogak, nyelv) figyeléséhez képest inkább annak tulajdonítható, hogy az arc redői kiemelik a szájmozgást, segítik az artikuláció pontosabb követését.

3. A vizuális lényegkiemelés

A szubjektív tesztek bizonyították, hogy az ellipszis méretét leíró ajakszélesség (a) és ajaknyílás (b) az ajkak láthatóságához hasonló felismerési eredményeket hozott. A képmomentumokból származtatható intenzitás faktor (k) - amely valójában a szájnyílás átlagos világossága - szolgált a magánhangzó és mássalhangzó felismerési kísérle-

teknél a nyelv és a fogak láthatóságának jellemzésére. A *k* intenzitás faktor a hátul képzett hangoknál a legkisebb (pl.: k, u). Közepes értékű, ha elül képzett hangoknál a nyelv látható (pl.: e, i). Legnagyobb a *k* értéke, ha a fogakat látjuk a szájnyílásban (pl.: s, cs).

A vizuális jellemzők kinyerésére kifejlesztett eljárások közös jellemzője, hogy a száj belső és külső kontúrjának követését követelik meg. [2, 5] Ezek a módszerek rendkívül számításigényesek, ezért lassúak. A számítási kapacitás növekedésével ez a probléma enyhül, de a legújabb módszerek sem elég megbízhatók. Az általam javasolt eljárás nem igényli a száj kontúrjának követését. A feldolgozásra kijelölt terület képi hasonlóságán alapul.

A számítógépek sebességének és tárkapacitásának további növekedésével a geometriai alapú rendszerek mellett előtérbe került a pixel bázisú feldolgozás. Ebben az esetben a száj és környezete, de akár az egész kép minden pontja részt vehet az elemzésben. A pixel bázisú feldolgozás előnye, hogy az ajkak környezetének feldolgozásával a szájmozgást kiemelő redőzet is a felismerés szolgálatába állítható. A teljes arc feldolgozása esetén a gesztusok figyelembe vételére is lehetőség nyílik. A geometriai alapú feldolgozás lehetővé teszi az artikuláció elemzését, a hangképzés statikus és dinamikus jellemzőinek mérését. A pixel alapú feldolgozás ezeket a lehetőségeket nem kínálja. Hátránya a pixel bázisú feldolgozásnak, hogy érzékenyebb a megvilágítás változásaira és személyfüggő felismeréshez használható.

A geometriai és pixel bázisú lényegkiemelés összehasonlítását csak igen kis méretű adatbázison, szó alapú elemzéssel végezték el. Feladatomnak tekintem a folyamatos audiovizuális beszédfelismerés szempontjait szem előtt tartó összehasonlító vizsgálat elvégzését. Az artikuláció dinamikus vizsgálatát csak a geometriai alapú elemzés szolgálja, ezért ennek kidolgozását elsőrendű fontosságúnak tartottam.

3. 1. A geometriai alapú vizuális lényegkiemelés

A jellegzetes képeken a szájsarkak elmozdulás vektorok vagy manuális segítség alapján kijelölhetők. Az ajkak méretét reprezentáló ajakszélesség (*a*) és ajaknyílás (*b*) ezek alapján meghatározható. Az ajkak belső és külső kontúrjainak követése igen nehézkes, ezért olyan módszer kifejlesztésére törekedtem, amely ezt nem igényli. A következő bekezdésben tárgyalt prototípus alakzatok jellemzőinek kialakításánál megengedhetőnek tartom a kézi beavatkozást, mivel csak a prototípus alakzatokra kell meghatározni őket. A szájnyílás belső területét a belső szájsarkak és az ajaknyílás felső illetve alsó széle közé rajzolt parabolák által határolt területtel közelítem, amelyre a *k* intenzitás faktor meghatározható.

Az adatbázisban szereplő videó anyag képkockáin kijelölt feldolgozandó terület jellegzetes ajakformáinak a különböző nyelvállásokat és a fogak eltérő láthatóságát figyelembe vevő prototípus alakzatok kiválasztásával artikulációs könyvtárt hoztam létre. Ezeken a képeken elvégeztem – a jellegzetes pontok esetleg manuális kijelölésével – a lényegkiemelést. Az adatbázis összes képének feldolgozásával meghatározva

képkockánként a hasonlóság mértékét, a legkevésbé hasonló alakzatokat felvettem az artikulációs könyvtárba. A műveletet addig ismételtem, amíg a legkevésbé hasonló alakzatok jellemzői bele nem simultak a környezetükbe. A prototípus alakzatok kiválogatása után 88 alakzat képviselte a jellegzetes képeket.

A módszer számos előnnyel jár az ismert eljárásokkal összehasonlítva:

- mérsékelt számításigényű, valós időben is elvégezhető a mai PC-ken
- a száj környezetét is figyelembe veszi, feldolgozási területe a geometriai alapú és a pixel bázisú módszerrel feldolgozott terület között helyezkedik el
- tetszőleges jellemzőket választhatunk a lényegkiemelésre, ezeket csak a kiválasztott képekre kell meghatározni, manuális támogatás is adható
- nem igényli a száj sem külső, sem belső kontúrjának meghatározását
- fekete-fehér képeken elvégezhető
- a vektorkvantáláson alapuló feldolgozás esetén közvetlen bemenetként szolgálhat

Hátránya, hogy beszélőfüggetlen feladathoz az artikulációs könyvtár bővítésére van szükség, ami a feldolgozási idő növekedéséhez vezet. A kutatás jelenlegi fázisában a beszélőfüggetlen audiovizuális beszédfelismerés nem tűzhető ki reális célként. Külön kutatási terület lehet az artikuláció személyfüggősége, a vizuális és az akusztikus jellemzők összefüggése.

3. 2. A pixel bázisú lényegkiemelés

A pixel bázisú lényegkiemelés az ajkak környezetének kijelölésével, rendszerint a képpontok számának decimációval történő redukálása után elvégzett transzformációt jelenti. A transzformációk a képsíkból a síkfrekvencia tartományba konvertálják a képeket. Erre a célra a diszkrét koszinusz transzformációt választottam, amelynek előnye a Fourier transzformációval szemben, hogy valós függvényeket valós függvényekbe konvertál. [4] A száj környezetének kijelölése a geometriai alapú feldolgozáshoz megtörtént, a sorok és oszlopok számát decimálással harmadolva 27x23 pontos képet kapunk. A transzformált jelből vízszintes és függőleges irányú síkfüggvényekből a 8-8 legkisebb síkfrekvenciájú 64 bázisfüggvény együtthatóiból válogattam a vizuális jellemzőket. A nagy síkfrekvenciájú komponensek a textúrát képviselik.

4. A felismerés alapegysége

Másik lényeges kérdés, amelyre a választ keresem cikkemben, hogy mit célszerű a gépi beszédfelismerés során a beszéd felismerendő nyelvi egységének tekinteni. Agglutináló nyelvek esetében a szóalapú feldolgozás folyamatos beszéd felismerésére nem alkalmas. Nincs általánosan elfogadott becslés a magyar nyelvben előforduló szóalakok számára vonatkozóan, annyi azonban bizonyos, hogy kezelhetetlen mennyiségről

van szó. Nyilvánvaló, hogy a fonéma szintű felismerés a hangok egymásra hatása miatt nem lehetséges. A szónál rövidebb, a fonémánál hosszabb alakzatokat célszerű választani felismerendő nyelvi egységként. A diád alapú és a Vicsi Klára [6] által javasolt félszótag alapú gépi beszédfelismerést kívánom összevetni. Összehasonlító elemzést végeztem a diád és félszótag alapú gépi beszédfelismerés tekintetében.

Vicsi Klára elemzései szerint a magyar nyelvű szövegek részleges lefedéséhez a félszótagokból kell a legszűkebb készletet figyelembe venni. A félszótag azért is ígéretes jelöltnek tűnik, mert általában hosszabb a diádnál, és hosszabb elemeket könnyebb megkülönböztetni egymástól. Hátránya, hogy a kezdő és záró félszótag csak a magánhangzónál illeszthető, az így képzett szótaghatárokon a hangok egymásra hatását nem tudja figyelembe venni.

A félszótagok egyik vetélytársa a diád lehet. A diád előnye, hogy mindkét végén illeszthető, figyelembe tudja venni a koartikulációs hatásokat. További előnye, hogy a teljes lefedéshez kevesebb elemre van szükség, mint a félszótagok esetében. Hátránya, hogy átlagos időtartama a félszótagokénál rövidebb.

4. 1. Az audiovizuális és az akusztikus adatbázis

Magyar nyelven nem áll rendelkezésre audiovizuális beszéd adatbázis, ezért a szerző bemondásával, házi videó berendezés és közönséges PC mikrofon felhasználásával felvett hang- és videó anyagon folyt a tanítás és tesztelés. A felvételek egyféle beállítással, az ülő helyzetben természetes fejmozgás mellett, speciális világítási előírások nélkül készültek. Az általánosabb alkalmazhatóság érdekében a színinformációt nem akartam felhasználni, ezért a képek feldolgozása a színes képek intenzitás képpé alakításával kezdődött. A felvétel körülményei normál irodai környezetnek felelnek meg. A hang a szobában működő, ventilátor zajt termelő PC-n került rögzítésre. Az utcáról beszűrődő zaj mellett a számítógép tápegysége által okozott zaj jelentős mértékű.

Az adatbázis felvételénél személyfüggő elemzést tűztem ki célul. Az audiovizuális adatbázis a számok és dátumok félszótag készletén alapul, a tanításra 486 szótag és 79 szó szolgált, a tesztelésre 35 szófüzért használtam. Az audiovizuális adatbázis a képkockák félképekre bontásával másodpercenként 50 képet tartalmaz, a szinkronitás végett az akusztikus jel is 20 ms-os lépésközzel kerül feldolgozásra. Az akusztikus lényegkiemelés 12 MFCC együtthatót tartalmaz, a mintavételi frekvencia 22 050 Hz.

Annak megbízhatóbb eldöntésére, hogy a felismerés alapegységeként a félszótag vagy a diád az alkalmasabb választás, egy nagyobb akusztikus adatbázist hoztam létre, saját bemondással 8000 szó szolgálta a tanítást, 1400 szó a tesztelést. A szavakban szereplő 121 kezdő félszótag kumulatív gyakorisága 70,2%, a 83 záró félszótag kumulatív gyakorisága 80,7%, a kiválasztott félszótagok a szótagok 60,1%-át, a szavak 32,6%-át fedték le. A statisztikai elemzés 1 996 589 szóból álló klasszikus és modern prózát dolgozott fel, amely 4 238 066 szótagot tartalmazott.

A diádokon alapuló elemzés ugyanezen az adatbázison történt. Az audiovizuális adatbázis a diádok teljes készletének mintegy 20%-át, az akusztikus adatbázis a diádok 50%-át fedte le.

5. A beszédfelismerési eredmények

A beszéd felismerését az audiovizuális adatbázison bimodális és unimodális gerjesztéssel is megvizsgáltam, mind félszótag, mind diád alapú elemzéssel.

5. 1. Az audiovizuális adatbázis felismerési eredményei

A geometriai bázisú vizuális lényegkiemelés félszótag alapú felismerési kísérletekben a hibák ötödét kiküszöbölte. A diád alapú beszédfelismerési kísérletben a geometriai bázisú lényegkiemelés vizuális kiegészítő jele alig volt képes javítani az akusztikus felismerési eredményeken. A pixel bázisú vizuális lényegkiemelés eredményeképpen a felismerési hibákat mintegy felére sikerült csökkenteni.

1. Táblázat. A helyes felismerés arányai az audiovizuális adatbázison

	Félszótag	Diád/Geometriai	Diád/Pixel
Akusztikus	69,1 %	88,7	88,7
Audiovizuális	75,9 %	89,8	94,2

5. 2. Az akusztikus adatbázis felismerési eredményei

Az akusztikus adatbázison végzett kísérletek is igen lényeges különbséget mutattak a félszótag és diád alapú felismerési eredményekben. A pusztán akusztikus gerjesztés esetében a félszótag alapú felismerés hibáit felezte a szótagok kapcsolódását kifejező *kötött félszótag* alapú felismerés bevezetése. A diád alapú felismerés hibáinak száma mintegy ötöde a kötött félszótag alapú felismerési hibáknak.

2. Táblázat. A helyes felismerés arányai az akusztikus adatbázison

	Félszótag	Kötött félszótag	Diád
Akusztikus	59,2 %	79,8 %	95,9 %

A kötött félszótag értelmezése: A koartikulációs hatások figyelembe vétele céljából a szótaghatárokon a félszótagokat diádokkal illesztve, (diád - kezdő félszótag - záró félszótag) sorozatok alkotják a láncot, a végén diáddal lezárva. A szótagok ilyen illesztése felére csökkentette a hibák számát, de a diád alapú felismerés eredményeit nem tudja megközelíteni.

6. Összefoglalás

Az audiovizuális beszédfelismerési kísérletek megmutatták, hogy a vizuális jel lényeges kiegészítő információval szolgálhat a beszéd felismeréséhez. A geometriai alapú vizuális lényegkiemelést sikerült az ajakkontúrok követése nélkül megoldani. A geometriai bázisú lényegkiemelés eredményei alapadatokat szolgáltattak az artikuláció dinamikus jellemzőinek elemzéséhez, amelyre egy vizuális beszédszintézis projekt épült. A pixel bázisú lényegkiemelés további javulást eredményezett. A vizsgálatok másik eredménye, hogy az ígéretesnek tűnő félszótag alapú felismerés nem képes a diád alapú felismerés eredményeit megközelíteni. A felismerés alapegysége kontextusfüggő, és ezt az elemeknek ki kell fejezniük.

Irodalomjegyzék

1. N. O. Bernsen: Multimodality in Language and Speech Systems – from Theory to Design Support Tool. in Multimodality in Language and Speech Systems. Kluwer Academic Publishers, Dordrecht/Boston/London 2002.
2. Luettin, J., Thacker, N.A., and Beet, S.W. (1996). Speechreading using shape and intensity information. Proc. International Conference on Spoken Language Processing, Philadelphia, PA, pp. 58–61.
3. Massaro, D.W. (1996). Bimodal speech perception: A progress report. In Stork, D.G. and Hennecke, M.E. (Eds.), Speechreading by Humans and Machines. Berlin, Germany: Springer, pp. 79–101.
4. Neti, C., Potamianos, G., Luettin, J., Matthews, I., Glotin, H., Vergyri, D., Sison, J., Mashari, A., and Zhou, J. (2000). Audio-Visual Speech Recognition. Final Workshop 2000 Report. Baltimore, MD: Center for Language and Speech Processing, The Johns Hopkins University.
5. Silsbee, P.L. and Bovik, A.C. (1996). Computer lipreading for improved accuracy in automatic speech recognition. IEEE Transactions on Speech and Audio Processing, 4(5):337–351.
6. Vicsi, K., Vigh, A. Text independent neural network/rule based hybrid, continuous speech recognition. EUROSPEECH'95. Madrid: pp. 2201-2204, 1995.

Megértést segítő részletező gépi névfelolvasás magyar nyelvre*

Fék Márk[1], Németh Géza[1], Olaszy Gábor[1,2], and Gordos Géza[1]

[1] BME Távközlési és Médiainformatikai Tanszék,
1117 Budapest, Magyar tudósok körútja 2.
{fek,nemeth,gordos}@tmit.bme.hu
[2] MTA Nyelvtudományi Intézete,
Kempelen Farkas Beszédkutató Laboratórium,
1399 Budapest, Benczúr u. 33, Pf. 701/518
olaszy@nytud.hu

Kivonat Az automatikus beszédválaszú számszerinti tudakozó névfelolvasó modulja olvassa be a telefonba a keresett előfizető nevét. A felovasott személy- vagy cégnév telefonon keresztüli érthetőségének növelésére, a szótagoláshoz hasonló, részletező felolvasási móddal egészítettük ki a rendszert. A szótagokra bontás felteszi, hogy magyar nyelvű szöveggel van dolgunk. Idegen írásmód esetén, illetve ha a telefonon keresztüli gépi hang érthetősége nem megfelelő, az adott szótag után a megértést segítő megjegyzéseket iktat be a rendszer. A cikk ismerteti a rendszer felépítését és a megvalósítás során felmerült problémákat.

Kulcsszavak: gépi beszéd, névfelolvasás, számszerinti tudakozó, automatikus szótagolás

1. Bevezetés

Az automatikus beszédválaszú számszerinti tudakozók a gépi beszédkeltés egyik fontos alkalmazási területét képviselik [Spiegel, 1993], [Nebbia et al., 1998], [Lundin, 1998]. Az első ilyen magyar nyelvű rendszert a Budapesti Műszaki és Gazdaságtudományi Egyetem, Távközlési és Médiainformatikai Tanszéke és a T-Mobile Magyarország Rt. fejlesztette ki közösen [Németh et al., 2003]. A rendszer a Profivox magyar nyelvű beszédszintetizátor [Olaszy et al., 2000] célirányos továbbfejlesztésével jött létre.

Ezekben a rendszerekben a felhasználó telefonon keresztül lép kapcsolatba az automatikus központtal, majd a nyomógombok segítségével megadja a keresett előfizető telefonszámát. A központ – ha adatbázisában megtalálta a telefonszámot, és az nem titkos – gépi beszéd segítségével beolvassa az ahhoz tartozó előfizető nevét és címét.

* A munkát a Nemzeti Kutatási és Fejlesztési Program, Alkalmazott beszédinformációs rendszerek projekt keretében támogatta

Gyakran előfordul, hogy az előfizető (magánszemély vagy cég) neve idegen, vagy rendhagyó helyesírású szavakat tartalmaz, így az ügyfélnek gondot jelenthet azok feljegyzése. Erre megoldást jelenthet az előfizető nevének betűzése (például: Techno='t, mint Tamás, e, mint Elemér,...'). A betűzés azonban lassú megoldás. Ennél gyorsabb, ha az idegen vagy rendhagyó helyesírású nevet szótagolva, és az egyes helyesírási sajátosságokat részletezve, olvassuk fel. Példaképp a 'technocity' nevet 'teh, céhával, no, kötőjel, ci, ti, ipszilonnal'-ként olvassa fel a gép.

A fenti szótagoló-részletező megoldás hasonló a magyar anyanyelvű beszélők által gyakran használt eljáráshoz, mely segítségével szótagokra bontva magyarázzák el egymásnak egy idegen eredetű szó helyesírását. Ez a betűzésnél természetesebb megoldás várhatóan kedveltebb lesz a felhasználók bizonyos körében.

2. A részletező felolvasást végző modul felépítése

A részletező felolvasási feladatot a név- és címfelolvasó szoftver ún. részletező modulja valósítja meg. Ennek blokkvázlata az 1. ábrán látható. A bemenet a feldolgozatlan szöveg, a kimenet a részletezett, hanglejtési és beszédszünet címkékkel ellátott szöveg. A feldolgozás lineáris, az egyes almodulok egymást követik. Az almodulok vagy lecserélnek egy adott szövegrészt, vagy kiegészítő megjegyzéseket szúrnak be. A beszúrt megjegyzések védelmet kapnak, így a további almodulok ezeket a szövegrészeket nem módosítják. A részfeladatokat szabályok és kivétel szótárak együttes alkalmazásával oldottuk meg. A kivételszótárakat memória- és sebesség-hatékony ternáris keresőfákkal valósítottuk meg [Bentley and Sedgewick, 1997].

3. Magánhangzók és mássalhangzók címkézése

A részletező felolvasáshoz a szövegben bejelöljük a szótaghatárokat. E művelet első lépése a szöveg karaktereinek felcímkézése magánhangzó, mássalhangzó, szám, illetve egyéb karakter kategóriákra. A magyar nyelvben fonológiai kategória a beszédhang hossza. A mássalhangzók esetében a hosszú hangot az írásban betűkettőzéssel jelöljük. Ebből kifolyólag a betűk környezetét is vizsgálni kell, hogy a hosszú hangot egyértelműen azonosíthassuk a részletező felolvasás számára.

A magánhangzók és a mássalhangzók betűképeit egy-egy adatbázis tartalmazza. A magánhangzók között szerepelnek cég és személynevekben előforduló idegen betűk (például az ä), a mássalhangzók között a két betűvel jelzettek (sz, zs, stb.) és ezek hosszú megfelelői (ssz, zzs, stb.) is. Emellett a nevekben gyakran előforduló idegen mássalhangzók, mint például az sch, szintén szerepelnek.

A régies írású nevekben előforduló cz betűkapcsolat kezelésére az esetek jó részét lefedő szabályt alkottunk. Ha mássalhangzó után jön a cz, akkor egy betűnek tekintjük, egyébként különálló c-nek és z-nek. Így például a 'Kótczy' név 'kót, ci, cézével, ipszilonnal'-ként részleteződik. Az olyan cz-t tartalmazó neveket, amelyek ez a szabály nem kezel, a későbbiekben működésbe lépő elválasztási

kivétel szótárban adtuk meg, így ezen szavak esetén a c és z között nem lesz elválasztás.

Külön problémát jelentett az y-os szavak kezelése, mivel az y-nak többféle jelentése is lehet. Kettős betűk második tagjaként (ny, ty, stb.) nem ejtjük külön, míg régies helyesírású és idegen szavakban i-ként (magánhangzó) vagy j-ként (mássalhangzó) ejtendő. Mivel egy ezeket kezelő szabály igen bonyolult, és semmiképpen nem lett volna teljes, ezért ezen eseteket kizárólag kivételszótárral kezeltük.

4. Szótagokra bontás

A szótaghatárokat a betűhatárok alapján állapítottuk meg. Ehhez a magyar nyelv szótagolási szabályát vettük alapul [Ferenc, 1992]. Eszerint minden magánhangzó külön szótagot alkot. A szótagot alkotó magánhangzó az ún. szótagmag. A szótag eleje a szótagmagot képező magánhangzót megelőző mássalhangzó (ha van ilyen), ellenkező esetben maga a szótagmagot képező magánhangzó. A szó elején lévő összes mássalhangzót a rá következő magánhangzó által alkotott szótaghoz rendeljük. A hosszú kettős betűket a magyar elválasztásnak megfelelően szótagoljuk. Eszerint a 'hosszú' és 'Illyés' nevek 'hosz, szú' és 'ily, lyés'-ként lesznek szótagolva. A program nem valósít meg morfológiai elemzést. Az összetett, -ság, -ség képzős, igekötős, kvázi igekötős és leg-gel kezdődő szavak helyes elválasztását a kivételszótár tartalmazza.

Az elválasztási kivétel-szótárakban szereplő szavak elválasztását a szótárban előírt pontokon módosítja a program. Itt egyrészt több ezer, az általános szabályoktól eltérő elválasztású szó (például összetett szavak, bizonyos cz-s szavak, stb.), másrészt a magyar szabályokhoz képest eltérő elválasztású idegen eredetű cég és személynevek szerepelnek. Ezeket több ezer cég és rendhagyó személynevet tartalmazó adatbázisok elemzése, illetve az elemek meghallgatása révén alakítottuk ki.

5. Rövidítések és írásjelek feloldása

A kizárólag nagy betűket tartalmazó szavak többnyire cégnevek, amelyeket bizonyos esetekben betűnként (pl. IBM), máskor egybe (pl. MATÁV) kell felolvasni. Ezek kezelésére egy kivételszótárral kombinált szabályrendszert alkottunk. A szabályrendszer alapvetően a szó hosszát és a benne szereplő magánhangzók számát figyeli. A csupa nagybetűt tartalmazó rövidítések elé a rendszer a 'csupanagybetűvel' megjegyzést szúrja be (pl. az IBM 'csupanagybetűvel, í, bé, em'-ként kerül felolvasásra).

A felolvasandó nevekben szereplő rövidítéseket (pl. 'Kft', 'Bt', stb.) a rövidítéseket feloldó modul teljes alakjukra ('káefté', 'bété') cseréli. Hasonlóan a szövegben szereplő írásjelek is lecserélésre kerülnek.

6. A helyes leírást segítő megjegyzések beillesztése

A részletező modul egyik fontos feladata, hogy segítse az ügyfelet a hallott név helyes lejegyzésében. Ezért részletező megjegyzéseket is fűzünk a szótagoláshoz. Itt például a w-s, y-os, ly-os, kettős mássalhangzót, hosszú magánhangzót, stb., tartalmazó szótagok kerülnek kommentálásra. Így például a 'Way Kft.' 'vaj, dupplavével, ipszilonnal, káefté'-ként lesz részletezve.

Az önmagukban álló árva betűkhöz helytelen lenne megjegyzés fűzése, ezért ezeket a következő modul egyszerűen a megfelelő szövegre cseréli. Így például egy önmagában álló 'w' 'duplavére' cserélődik.

A még nem lecserélt idegen betűk, tipikusan a q-betűk, egy nekik megfelelő helyettesítő szövegre cserélődnek. Például a 'Quarz' szó 'kú betű, u, arz'-ra cserélődik. Ez biztosítja, hogy a q-betűs helyesírás egyértelmű legyen a felhasználó számára. A q-betű kezelését, ugyanis nem lehetett más idegen betűk (pl. y, w, stb.) kezelésekor követett módszerhez hasonlóan megoldani.

A meghallgatásos tesztek során egyes szótagokban nehéz volt eldönteni, hogy azok hosszú, vagy rövid magánhangzót tartalmaznak-e. Ezért a rendszer hosszú magánhangzók esetén is kiegészítő megjegyzést szúr be (pl. az Ír Bt. 'ír, hosszúível, bété'-ként kerül felolvasásra).

7. Prozódiai címkék beillesztése

A szótagokra tagolt, részletezett szöveg felolvasása nem igényli bonyolult dallamenet alkalmazását. Meghallgatásos vizsgálataink alapján elegendőnek bizonyult a bemondás utolsó szótagjában levinni a hangmagasságot. A beszédszintetizátor dallammenetét a szövegbe helyezett speciális vezérlő jelzésekkel lehet vezérelni. A részletező modul egy ilyen speciális jelzést helyez az utolsónak kimondásra kerülő szótag elé.

A megfelelő ritmika kialakításának érdekében eltérő hosszúságú szüneteket iktattunk be a szótagok, illetve a hozzájuk tartozó megjegyzések közé. Meghallgatásos tesztek során több változat közül olyan értékeket választottunk, amelyekkel a legjobb volt a felolvasott szöveg érthetősége. Ezen szünetek is speciális jelzésekként (pl. [:pause 500]) kerülnek a szövegbe.

A meghallgatásos tesztek során arra is fényderült, hogy még hosszabb szünetek alkalmazása esetén is nehézséget okozhat a felolvasandó nevekben esetlegesen szereplő szóhatárok elkülönítése. Emiatt a szóhatár hollétének markáns jelölésére azt 'szóköz'-ként külön be is mondja a rendszer. Például a Balta Bán Kft 'bal, ta, szóköz, bán, káefté'-ként kerül bemondásra. Az ennek megfelelő szövegrészt a szünetekkel egyszerre illesztjük be.

A kimondott szöveg érthetőségét tovább növelendő, az egyes szótagokat lassabban, míg a hozzájuk fűzött megjegyzéseket gyorsabban olvassa a rendszert. Ezen sebesség-beállításokat jelző parancsoknak az elhelyezésére is itt kerül sor.

8. Összefoglalás

A fent leírt módon előkészített szöveget a név-és címfelolvasó szintetizátor modulja szótagolva, részletezve, megfelelő szünetekkel, intonációval és sebességgel olvassa fel.

A rendszert több-ezres személy és cégnév adatbázisokon végzett meghallgatásos tesztekkel ellenőriztük, illetve folyamatosan fejlesztettük. Az adatbázisokat egyrészt a velünk együttműködő távközlési szolgáltatótól (T-Mobile Magyarország Rt.) kaptuk. Ezek azonban személyiségi és egyéb jogvédelmi szempontok miatt érthetően korlátozottak voltak, így a teszteléshez más forrásból származó cég és személynév adatbázisokat is felhasználtunk.

A rendszer 2004 februárja óta érhető el a T-Mobile 1230-as ügyfélszolgálati számán keresztül. A 2004. szeptemberéig összegyűlt forgalmi adatok alapján a lekérdezések 63.5%-a egyszerű névfelolvasás, 17,5%-a részletező felolvasás, 19%-a pedig betűzés volt.

Hivatkozások

[Bentley and Sedgewick, 1997] Bentley, J. L. and Sedgewick, R. (1997). Fast algorithms for sorting and searching strings. In *SODA: ACM-SIAM Symposium on Discrete Algorithms (A Conference on Theoretical and Experimental Analysis of Discrete Algorithms)*.

[Ferenc, 1992] Ferenc, K., editor (1992). *Struktúrális magyar nyelvtan, 2. kötet – Fonológia, 4.4. fejezet: Szótagolási szabályok.* Akadémiai Kiadó, Budapest.

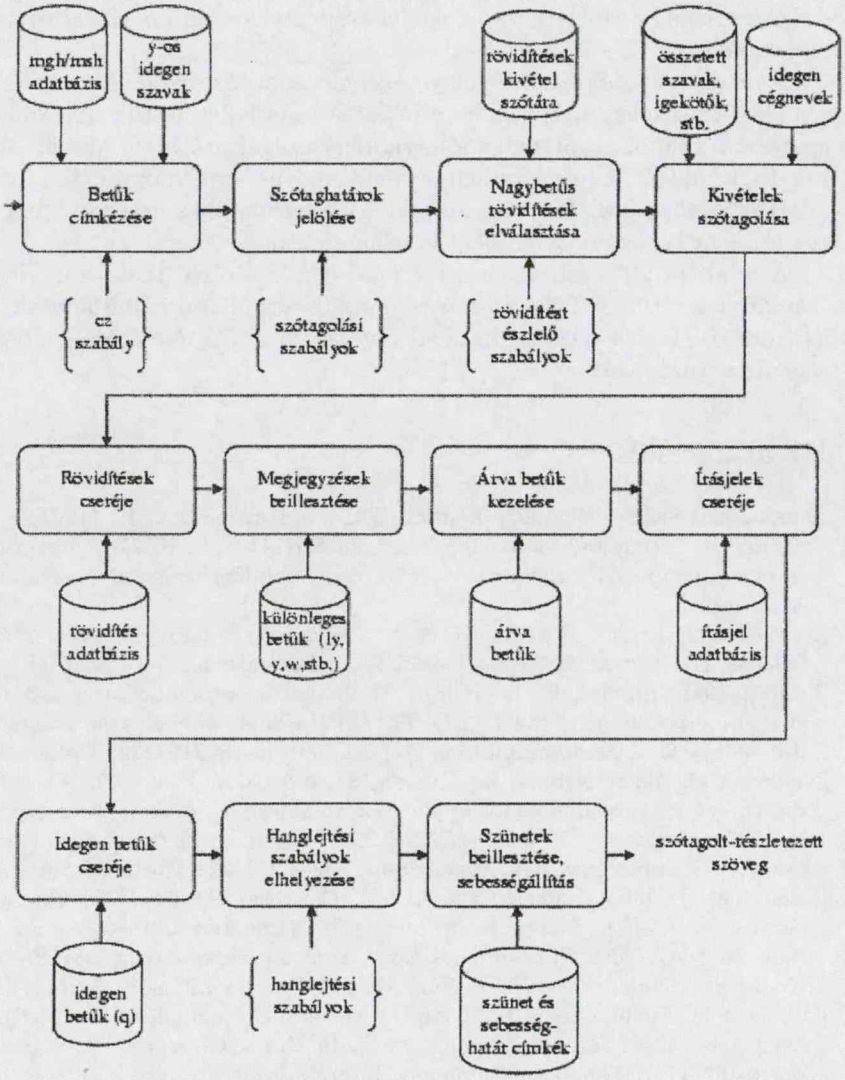
[Lundin, 1998] Lundin, F. J. (1998). The swedish automatic reverse directory service. In *Proceedings of the IVTTA'98, IEEE-ESCA Workshop on Interactive Voice Technology for Telecommunication Applications*, pages 219–222, Torino, Italy.

[Nebbia et al., 1998] Nebbia, L., Quazza, S., and Salza, P. L. (1998). A specialized speech synthesis technique for application to automatic reverse directory service. In *Proceedings of the IVTTA'98, IEEE-ESCA Workshop on Interactive Voice Technology for Telecommunication Applications*, pages 223–228, Torino, Italy.

[Németh et al., 2003] Németh, G., Zainkó, C., Kiss, G., Fék, M., Olaszy, G., and Gordos, G. (2003). Language processing for name and address reading in Hungarian. In *2003 IEEE International Conference on Natural Language Processing and Knowledge Engineering (IEEE NLP-KE 2003)*, pages 238–243, Beijing, China.

[Olaszy et al., 2000] Olaszy, G., Kiss, G., Németh, G., and Olaszi, P. (2000). Profivox: a legkorszerűbb hazai beszédszintetizátor. In Mária, G., editor, *Beszédkutatás 2000*, pages 167–179. MTA Nyelvtudományi Intézete.

[Spiegel, 1993] Spiegel, M. F. (1993). Coping with telephone directories that were never intended for synthesis applications. In *Proceedings of the ESCA-NATO/RSG 10 Tutorial and Workshop on Applications of Speech Technology*, pages 19–22, Lautrach, Germany.



1. ábra. A névfelolvasó részletező moduljának felépítése.

Beszédfelismerő modellépítési kísérletek akusztikai, fonetikai szinten, kórházi leletező beszédfelismerő kifejlesztése céljából

Velkei Szabolcs, Vicsi Klára

BME Távközlési és Médiainformatikai Tanszék, Beszédakusztikai Laboratórium

1117 Budapest, Magyar tudósok krt. 2.

E-mail: vicsi@tmit.bme.hu, velkei@tmit.bme.hu

Kivonat: Cikkünkben a Beszédakusztikai Laboratóriumban kifejlesztett HMM alapú beszédfelismerő rendszert, a rendszer optimalizálását mutatjuk be, és a felismerési eredményeinket összehasonlítjuk a széles körben elterjedt Hidden Markov Model Toolkit (HTK) rendszerrel kapott eredményekkel. A kutatás folyamatos, most az első évben a fonetikai felismerési szintet fejlesztettük ki, optimalizáltuk az akusztikai és a fonetikai szinteket. Az összehasonlító kísérletek azt mutatták, hogy az általunk kifejlesztett beszédfelismerő eljárás akusztikai szintű optimalizálásával valamint az akusztikai–fonetikai modellek optimalizálásával növelni tudtuk a felismerési pontosságot, és gyorsítani tudtuk a feldolgozást.

1 Bevezetés

Munkánk célkitűzése a Beszédakusztikai Laboratóriumban egy középszótáras, általános magyar nyelvű, folyamatos beszédfelismerési technológia kidolgozása, valamint egy ahhoz tartozó nyelvi modell elkészítése, amelynek segítségével a rendszer meghatározott kötött témában, közepes szótárméret alapján működik, rögzített nyelvtani keretek között, kis-zajú környezetben.

A 2004 év elején kezdődött, és 3 évig tartó project keretén belül a Laboratóriumban új megoldásokat dolgozunk ki az akusztikai előfeldolgozásban, a statisztikai modellépítésben valamint fonetikai, fonológiai és morféma nyelvi szinteket vonunk be a felismerési folyamatba. A felismerési kísérletekhez HMM alapú saját fejlesztésű beszédfelismerő rendszert állítottunk össze, a szokásos Hidden Markov Model Toolkit (HTK) [6] rendszert összehasonlításra használtuk. Az első évben a fonetikai felismerési szint feldolgozását optimalizáltuk. Cikkünkben erről az optimalizálási folyamatról és a kifejlesztett MKBP 0.8 felismerőről számolunk be.

2 Akusztikai előfeldolgozási eljárás optimalizálása

Akusztikai előfeldolgozási eljárásra a beszédfelismerés immár több évtizedes fejlődése alatt számos eljárás született, melyek közül ma az alábbiak a legismertebbek [1]: szűrősoros elemzés (BFFP), lineáris predikció analízis (LP), perceptuális lineáris predikció (PLP), kepsztrális együtthatók vizsgálata (MFCC), spektrális torzításon alapuló rendszerek (SDM).

A mai legsikeresebb felismerők előfeldolgozási rendszere mell-kepsztrum (MFCC) vizsgálatot végez: valamilyen hallásmodell alapján (Mel, Bark) [4,8] kiszámított szűrők sorozata szolgáltatja a bemeneti vektort (szűrősoros elemzés):

$$\text{Pl: Bark szűrő: } 10\lg L(x) = 15.8 + 7.5(x+0.5) - \sqrt{17.5(1+(0.5x)^2)} \tag{1}$$

$$\text{Mel-szűrő: } \mathrm{Mel(f)} = 2595\log_{10}\left(1+\frac{f}{700}\right) \text{ háromszögszűrő} \tag{2}$$

(f: frekvencia)

Az információ jobb kinyeréséhez kepsztrális együtthatókat képzünk (MFCC):

$$c_i = \sqrt{\frac{2}{N}\sum_{j=1}^{N} m_j \cos\left(\frac{\pi i}{N}(j-0.5)\right)} \tag{3}$$

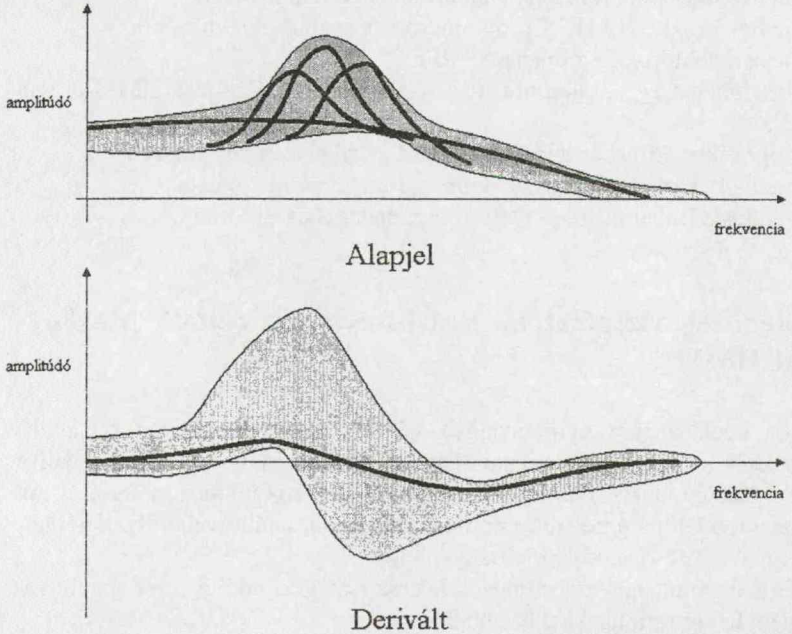
(c: kepsztrális együtthatók, m: szűrők energiái)

A kepsztrális együtthatók információtartalma igen magas, azonban az emberi szem számára nem hordoz jól felismerhető jegyeket. Ennek ellenére (vagy éppen ezért) az MFCC nagy népszerűségnek örvend, az egyik legtöbbet alkalmazott módszer. Ennek okai a következők lehetnek: Kizárólag a Mel- illetve Bark szűrősorokkal végzett akusztikai előfeldolgozáskor, a szűrősorok kimenetei ugyan hasonlóak a különböző ejtésekben, de egymáshoz viszonyítva spektrálisan el vannak tolódva (nagy a varianciájuk), a szűrősorral betanított Markov-modellekkel a betanítástól teljesen eltérő jeleket is lehet ismerni. Ebből kifolyólag a szűrősorral képzett eredmény vizuálisan ugyan jól meghatározott képet ad (emelkedés, süllyedés, maximumok), de mindenképpen kevés az információ. A (3.) képletben közölt módszer csökkenti a varianciát.

Felvetődött a kérdés, hogy miért ne készíthetnénk olyan akusztikai elemzést, melyben az eredeti szűrősoros adatok frekvenciatérbeli deriváltjai kapnak szerepet. A deriváltak a szomszédos szűrők különbségeként nyerhetők ki. Ez az eljárás nem teljesen idegen a szakirodalomban, de ma a gyakorlatban nem használatos. A frekvencia deriváláskor létrejött adatkiemelést ábrázolja a 2. ábra.

A felső szürke kitöltéssel stilizált eloszlás az adatok Bark szűrősoron való áteresztésekor keletkezett. Az eloszlást létrehozó bemeneti adatok egy része zöld színnel ki lett emelve. A képen jól látható, hogy a modell a piros függvényhez tartozó bemeneti vektorokat is nagy valószínűséggel képes felismerni, pedig a betanító minták ettől szignifikánsan különböztek. Az alsó ábrán az eddigi Bark szűrősor frekvenciabeli

deriváltjaiból képzett modell bemeneti vektor-eloszlása látható. Megfigyelhető, hogy mindenképpen szükséges egy minimális egyezés a derivált kilengésénél. Ezáltal a csúcs nem tűnt el, mint az előbbi esetben, és a modell csak olyan eseteket ismer fel,



2. ábra: Bark szűrősor és Bark szűrősor deriváltjainak eloszlása

amikor a vizsgált kritikus helyen valóban van energiamaximum. Továbbá, míg az igen népszerű MFCC számításigénye N^2-el arányos, az új módszeré csak N-el.

Találati százalékos eredmények 20 kHz mintavételezési frekvencia mellett 3 és 5 állapotú, fonéma alapú diszkrét Markov-modellekkel az 1. táblázatban találhatók.

1. Táblázat. Felismerési találatok frekvenciatérbeli és időbeli deriválás esetén.

Találat	Bemeneti vektor	Állapotszám
44.36%	23 Bark szűrő	3 állapot
46.1%	23 Bark szűrő	5 állapot
46.15%	23 Bark szűrő + 23 Bark időbeni derivált	3 állapot
64.38%	23 Bark frekvenciatérbeli derivált	3 állapot
70.16%	23 Bark frekv. derivált + 23 időbeni derivált	3 állapot
72.32%	23 Bark frekv. derivált + 23 időbeni derivált	5 állapot
71.38%	13 MFCC+13 Delta+13 Acc. komponens	5 állapot

A Betanítás Babel magyar nyelvű beszédadatbázissal történt [5].

Referencia-felismerő

A referencia-felismerő ajánlását a COST249 később annak lejárása utána COST278-as munkacsoport dolgozta ki [7] amelynek paraméterei az alábbiak:

- a modellek akusztika fonémák halmaza a megfelelő nyelven,
- a modellek készítése a HTK programcsomag segítségével történik,
- a bemeneti vektorok 39 dimenziós MFCC,
- minden fonéma egy 3 állapotú 'ballról-jobbra haladó' típusú HMM-el van modellezve,
- az ortografikus átírást és a kiejtési szótárt az adatbázis tartalmazza,
- diagonális kovariancia-mátrixú Gauss paraméterek használata.
- A betanítás a Babel magyar nyelvű beszédadatbázissal történt.

3 Modellépítési vizsgálatok, kvázi-folytonos rejtett Markov-modellek (QCHMM)

A 80-as évek végére nyilvánvalóvá vált, hogy diszkrét (vektorkvantált) Markov-modellekkel a felismerés nem javítható tovább, ezért ki kellett dolgozni a folyamatos valószínűségi mezőkre értelmezett modelleket, azok tanítási módszerét. Az alapelv ugyanaz: a modell paramétereire optimális értékeket találni valamilyen iterációs algoritmus segítségével. A módszer hátránya, hogy

- nagybonyolultságú algoritmusok jelennek meg az eddigi négy alapműveletet igénylő algoritmusokkal szemben,
- új probléma merül fel: a gaussi változók szórásainak figyelése és esetleges módosítása,
- a robusztus matematikai módszer sok időbe kerül – lassú programot eredményez.

Nyilvánvalóan ez nem probléma amennyiben lehetőség van nagyteljesítményű vektorszámítógépek használatára, de ennek hiányában kifejlesztettünk egy kvázi-folytonos rejtett Markov-modelleket (QCHMM) használó programot, mely a fenti két vetélytárs jó tulajdonságait igyekszik ötvözni, nevezetesen a nagyobb pontosságot a kisebb futási idővel.

Ezen problémák megoldásához a kvázi-folytonosság a kulcs, mely a következőt jelenti: a meglévő N felbontású diszkrét mezővel nem a tanítóanyag mintáinak eloszlását kell maximális hűséggel visszaadni, hanem azok alapján egy becslést adni a folytonos valószínűségi mezőre. Ehhez egy simítási algoritmusra van szükség, amelyet egyszer vagy többször végigfuttatva az eddig betanított modellen a modell tanítóanyagra való adaptálódását lehet csökkenteni (megszüntetni). A használt algoritmus egy paraméterezhető simítófüggvény (továbbiakban blur), melynek a következőket lehet megszabni:

- az élsimító mátrix (1 dimenzió miatt jelen esetben csak vektor) méretét,
- a mátrix elemeinek értékét, ahol minden sorban összértékben 1-nek kell lennie (a teljes valószínűségi mező mindig 100%)

Megoldandó problémák:

- kvantálási lépcsők számának optimális megválasztása,
- megfelelő súlyozású simítófüggvény megválasztása,
- a tartomány minél jobb eseménytérbeli kihasználtsága.

Az optimális kvantálási együttható, valamint az ehhez tartozó simítófüggvény mérési úton lett kiszámítva. Többféle adatbázisrész felismerési aránya nagyszámú véletlenszerűen választott paraméterezésű felismerő kimenetén össze lett vetve, majd a statisztikailag így megismert kétváltozós probléma maximumkereséssel lett megoldva.

Végeredményben tehát a laboratóriumban kifejlesztett fonémaszintű felismerőnk, a továbbiakban MKBF 0.8, 16 kHz mintavételezésű, 17 Bark frekvenciatérbeli derivált + 17 időbeni derivált + 17 időbeni második derivált + energia bemeneti jelvektor mellett, 4-5 állapotú kvázi-folytonos, 24 lépcsős, rejtett Markov-modellekkel (QCHMM) fonéma, illetve trifon alappal dolgozik.

Kulcsfontosságú lépés az eredmények összehasonlítása más eljárások eredményeivel. Ennek elvégzéséhez a már megismert HTK 3.2 Markov-modellre épülő fejlesztő szoftver nyújtott segítséget, mely egy teljes körű beszédfelismeréshez kialakított alkalmazás. A teszthez a Babel adatbázis anyaga került feldolgozásra a következő módosítással:

- 20 kHz mintavételezés helyett 8 kHz-es újra mintavételezés.

A HTK a szerzők ajánlása szerinti legjobb paraméterezéssel tanulta meg a modelleket:

- előszűrés: $s_n' = s_n - k \cdot s_{n-1}$, ahol $k=0.97$,
- Hamming ablak: $s_n' = \left\{0.54 - 0.46\cos\left(\frac{2\pi(n-1)}{N-1}\right)\right\} s_n$
- 1,2,4,8 illetve 16 Gauss-keverék használata az eloszlás modellezéséhez
- maximum 20 iterációs mélység
- 13db MFCC és ezek első és második időbeli deriváltjai (összesen 39 elem)
- a paraméterek részletesebb megismerése a HTK könyvéből [6] lehetséges.

Az MKBF 0.8 felismerőnk felismerési eredményeinek összehasonlítása a HTK 3.2 alapú referenciafelismerővel a 2. táblázatban találhatók. A táblázat alapján kivehető legfontosabb felfedezés azonban az, hogy a QCHM modellek esetében a felismerési arány gyakorlatilag nem romlott a lépcsők számának drasztikus csökkentésével.

2. táblázat: Az MKBF 0.8 sajátfejlesztésű felismerő felismerési eredményeinek összehasonlítása a HTK 3.2 alapú referenciafelismerővel

HTK 3.2	
Paraméterszám	találati arány
1 Gauss	52.3%
2 Gauss	58.8%
4 Gauss	61.2%
8 Gauss	62.3%

MKBF 0.8	
Paraméterszám	találati arány
6 lépcső/ 0 blur	67.61%
12 lépcső/ 0 blur	64.98%**
24 lépcső/ 4 blur	68.188%
48 lépcső/6 blur	69.55%

32 Gauss használatát a HTK a kevés mintaanyagra hivatkozva elvetette

A laboratóriumban a 80-as években végzett kutatások alapján kimutatták, hogy az emberi fül dinamikában csak meglepően nagy intenzitás-különbségeket (5-6 dB) képes észrevenni. A beszéd intenzitásának tartománya körülbelül 35 dB, így 6-7 lépcsővel jellemezhető az emberi fül 'kvantálása'. A szakirodalomban található SISI-teszt [2], erről a jelenségről tanúskodik.

Tehát a felismerésnél kapott eredmények összhangban vannak a dinamikára vonatkozó szubjektív akusztikai vizsgálatokkal. Az is megfigyelhető, hogy ugyanaz a lépcső/simítás≈7 arány mellett lettek a felismerések a legjobbak. Ez érthető, hiszen a simítás durva közelítéssel megfeleltethető a kvantálási lépcsők csökkentésének is.

Kiértékelési módszerek

Az előforduló hibákat értelmezni kell, erre jó módszer lenne, ha minden esetben meg lehetne mutatni, hogy a felismerő mikor milyen típusú hibát vétett. A hibák típusai a következők lehetnek:

- csere: valamely szimbólum tévesztése egy másikkal,
- beszúrás: két szimbólum közé újabb szimbólum(ok) kerül(nek),
- törlés: valamely szimbólum hiányzik.

Amennyiben a felismerő kimenetén a modellek sorozata található meg, a fenti módszer alkalmazható, ellenben a program a kimeneten adhat egyéb információkat is: például időzítési adatokat. Ebben az esetben lehetőség van utólagos módosításra, mivel a statisztikai alapon gyanús eredményeket szűrni lehet. Ilyen megfontolásból legyen a hiba definíciója a következő:

- Ha egy frame alatt a felismerő kimenetén különböző modellt detektálunk, mint a bemeneti szimbólumhoz tartozó modell, akkor az hibának tekintendő.

Ezzel a módszerrel egy könnyen algoritmizálható hibadefinícióhoz jutottunk, amelynél a hibapontszám továbbra is lineáris kapcsolatban áll a szimbólumhibákkal, de ez esetben hibapont adódik az offszethibákhoz is. A kiértékelésről és a modellek betanításáról részletesebb információkat közöl a szerző TDK dolgozata [3] .

4 Fonetikai modellek

A felismerésben használt fonetikai modellek háromféle típusból lettek összeválogatva:

- a modellek magját a magyar nyelvben használt fonémák rövid-hosszú párjai alkotják, illetve csekély számú fonémánál allofonok lettek megkülönböztetve, (zöngés zöngétlen h, stb.)
- a beszéddetektáláshoz használható beszéd-nem beszéd modell háromelemű: csend, rezonáns és zörej osztályba sorolja a hangokat,
- míg végül a pontosabb felismerés érdekében trifón modelleket alkalmazunk.

Trifón modellek esetén meg lettek különböztetve valódi trifón elemek és osztályozott fonémák. Második esetben ugyanis azokat a fonémákat, amelyekhez a tanító adatbázis nem tartalmazott elegendő információt valamennyi szóba jövő trifón betanításához, ott a kezdő és záró fonémák helyett fonémaosztályokat alkalmaztunk. A trifón modellek használatával a beszéd felismerése előreláthatólag 5-10% fog javulni.

5 Összefoglalás

Az összehasonlító kísérletek azt mutatták, hogy az általunk kifejlesztett beszédfelismerő eljárás (MKBF 0.8) akusztikai szintű optimalizálásával valamint az akusztikai–fonetikai modellek optimalizálásával növelni tudtuk a felismerési pontosságot, és gyorsítani tudtuk a feldolgozást. Természetesen a további szintek igen jelentős mértékben javítani fogják a felismerő pontosságát, de egy jobb kiindulással biztosabb az eredmény.

Köszönetnyilvánítás

A kutatás az OTKA T 046487 ELE és az IKTA 00056 pályázatok keretén belül készül.

Bibliográfia

1. Bechetti, C., and Prina Ricotti, L.: Speech Recognition (John Wiley and Sons LTD 1999.)
2. Pytel, J.: Audológia (Victoria kft., 1996)
3. Velkei, Sz.: Rejtett Markov-modell elméleti és gyakorlati optimalizálása folyamatos beszédfelismeréshez. TDK dolgozat. BME 2004, pp. 29-40
4. Vicsi, K. Matilla, M. and Berényi, P. (1990). Continuous Speech Segmentation Using Different Methods, Acustica, Vol. 71, 152-156. Video Voice, Micro Video, 210 Collingwood, Suite 100. PO Box 7357 Ann Arbor, MI 48107
5. Vicsi, K., A. Vig: Az első magyar nyelvű beszédadatbázis, Beszédkutatás'98, Tanulmányok az elméleti és alkalmazott fonetika köréből. MTA Nyelvtudományi Intézet, Budapest, pp. 163-178, 1998.
6. Young, S. et al.: The THK Book for HTK Version 3.2 (Cambridge University Engineering Department 2001-2002, http://htk.eng.cam.ac.uk/docs/docs.shtml)

7. Zgank, A., Kaịciịc, Z., Diehl, F., Vicsi, K., Szaszak, G., Juhar, J., Lihan, S., 2004. The COST 278 MASPER iniative - crosslingual speech recognition with large telephone
database. Proc. LREC 2004 Lisbon, Portugal.
8. Zwicker, E. and Terhardt, E. 1980. Analytical expressions for band rate and critical bandwidth as a function of frequency, J. Soc. Am. Vol. 68, 1523.

Beszédadatbázis irodai számítógép-felhasználói környezetben

Vicsi Klára [1], Kocsor András [2], Teleki Csaba[1], Tóth László[2]

[1] BME Távközlési és Médiainformatikai Tanszék, Beszédakusztikai Kutatólaboratórium, Sztoczek u. 2., 1111 Budapest, Magyarország {vicsi, teleki}@tmit.bme.hu
ttp://alpha.tmit.bme.hu/speech
[2] MTA-SZTE Mesterséges Intelligencia Kutatócsoport, Aradi vértanuk tere 1., 6720 Szeged, Magyarország
{kocsor, tothl}@inf.u-szeged.hu

Kivonat: Az előadásban bemutatjuk az új olvasott szöveget tartalmazó Magyar Referencia Beszédadatbázist, amelyet általános felhasználói környezetben, irodákban, laboratóriumokban, lakásokban rögzítettünk, összefoglaljuk az adatbázis létrehozásának akusztikai, nyelvi feldolgozási módszereit, ismertetjük az adatbázis pontos jellemzőit. Az adatbázis 332 beszélő olvasott szövegét tartalmazó hanganyag, beszélőnként 12 mondat és 12 szó speciális fonetikai elvárásoknak megfelelően összeállítva került bemondásra. A felvételek többféle mikrofonnal, hangkártyával és személyi számítógéppel készültek.

1 Bevezetés

Célunk egy általános felhasználású, irodai, otthoni környezetben olvasott folyamatos szöveget tartalmazó, beszédadatbázis létrehozása és akusztikai, valamint nyelvi feldolgozása, amely alkalmas PC-s beszédfelismerők betanítására, tesztelésére. A létrehozott beszéd adatbázist *Magyar Referencia Beszédadatbázis*nak *(MRBA)* neveztük el. A BME TMIT Beszédakusztikai Laboratóriuma a szegedi SZTE Informatikai Tanszékcsoporttal együttműködve hozza létre ezt az adatbázist. A BME TMIT Beszédakusztikai Laboratóriuma végezte el a beszédadatbázis szöveganyagának megtervezését, az előfeldolgozást és annotálást, az automatikus betű – fonéma átírást, a szegedi SZTE Informatikai Tanszékcsoport a kézi szegmentálást végezte, a felvételek készítése megosztva történt.

1.1 A beszédadatbázis szöveganyagának megtervezése

A létrehozandó adatbázis számos különböző típusú beszédfelismerő betanítására és tesztelésére kell hogy lehetőséget adjon. Mivel a felismerés alapja ma már szavaknál kisebb egység, fonéma, difon, trifon, stb., olyan folyamatos szöveg összeállítására van szükség, ahol ezek az elemek elegendően sokszor fordulnak elő. A beszédfelismerési célokra rögzítendő beszédnek a lehető legjobban kell fednie a beszélt magyar nyelv

sajátosságait. Mivel a felvett szöveg minden másodperce sok munkát von maga után, a szöveganyagnak minél rövidebbnek is kell lennie. Ezért alapos statisztikai vizsgálatokra van szükség a fonémák, difonok, trifonok szintjén egyaránt. Ennek alapján kell összeállítani olyan folyamatos szöveget, ahol a leggyakoribb bi- és trifonok megfelelő mennyiségben állnak rendelkezésre.

A szöveget fonémák sorozatára alakítottuk egy speciális algoritmus segítségével. Fonotipikus fonetikai átírást alkalmaztunk (Vicsi 2001, Fourcin, A.J. and Dolmazon, J-M.1991), vagyis a karakterek átírását a nyelv fonetikai szabályainak alapján végeztuk el úgy, hogy a szövegkörnyezetet is figyelembe vettük (pl. koartikuláció, hasonulás).

Az ortografikus karakterek fonetikai átírásához SAMPA fonetikai jelölést használtunk (Vicsi, 2000). A fonémaátírás után következett a statisztikai vizsgálat. A fonéma, bifon és trifon megoszlást vizsgáltuk az eredeti 1,6 MB méretű szövegadatbázison.

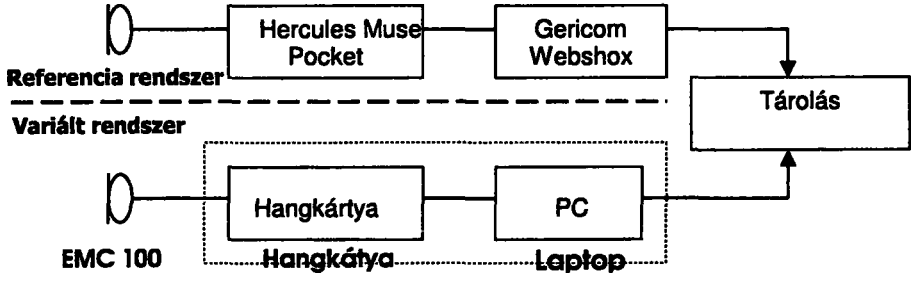
Egy bemondó 12 különböző mondatot és 12 különböző, a mondatoktól független, szót olvas fel. A teljes szöveganyag 166 bemondásra készült el azért, hogy szöveganyag 2-szer kerülhessen ismétlésre a 332 bemondás során. Így a teljes szöveganyag 12 x 166 azaz összesen 1992 db különböző mondatból és 12 x 166, azaz 1992 db különböző, a mondatoktól független szóból áll.

A mondatok kiválasztása a következő szempontok szerint történt: a fonémastatisztika alapján az 1%-nál gyakoribb fonémákat legalább egy példányban tartalmaznia kellett minden csoport valamelyik mondatának.

A teljes szöveg és a válogatott szöveg bifon statisztikáinak alapján a leggyakoribb 98.8% bifon mindegyike szerepel mindkét anyagban, és csak itt kezdenek el hiányozni a válogatott szövegből a bifonok.

2 A beszédadatbázis rögzítése és feldolgozása

A beszédadatbázis felvételeit különböző helyszíneken: zajos, kevésbé zajos iroda



1. ábra Az MRBA adatbázis felvételi elrendezése

helyiségekben, laborokban, otthonokban vettük fel. A felvételeknél szinkronban két különböző rendszerrel dolgoztunk. Az egyik az ún. *referenciarendszer*, ahol jó minőségű, közelbeszélő kondenzátor mikrofont (Monacor EMC 100), jó technikai paraméterekkel rendelkező hangkártyát (Hercules Muse Pocket USB 5.1), valamint egy adott Gericom Webshox típusú laptopot használtunk. A másik rendszernél az ún.

variált rendszer, különböző, jobb, kevésbé jó mikrofonokat, hangkártyákat, PC-ket használtunk, a lehető legnagyobb variáltsággal. A felvételi elrendezést az 1. ábra mutatja. Felvételeknél a bemondók, a PC-k, a hangkártyák, a mikrofonok és a környezet minél nagyobb variáltságára törekedtünk.

2.1 A beszélők demográfiai adatai

Régiók és dialektusok: A felvételeket Magyarország négy különböző tájegységében lévő városban rögzítettük: Budapesten, Szegeden, Győrben és Miskolcon. A felvételek során a beszélők születési helyét és jelenlegi lakhelyét jegyeztük fel.
Táblázat 3. A beszélők életkor és neme szerinti megoszlása

Korcsoport	Férfi beszélők	Női beszélők
16 év alatt	0,9 %	3,3 %
16 – 30 év	46,1 %	27,7 %
31 – 45 év	5,7 %	6 %
46 – 60 év	3,9 %	5,1 %
60 év felett	0,9 %	0,4 %
Összesen	*57,5 %*	*42,5 %*

2.2 Előfeldolgozás és annotálás

Az előfeldolgozás a felvételek lehallgatásából, ellenőrzéséből és esetleg feldarabolásából, esetünkben a két csatorna szinkronizálásából áll.
Az annotálás annyit jelent, hogy minden hangfájl mellé egy címkefájlt készítünk, amely különféle információkat tartalmaz a hangfájl paramétereivel és tartalmával kapcsolatban: az elhangzott szöveg ortografikus lejegyzését, hibás kiejtést, nem érthető szavakat, szótöredékeket, a beszélő nem beszédből származó hangjait, környezeti zajokat, stb. (Wells, J. 2001). Az alábbi ábrában egy ilyen cimkefájlt prezentálunk.

2.3 Fonotipikus automatikus betű-fonéma átírás, szóhatár megadás

A bemondott szöveg betűit átírtuk úgy, hogy figyelembe vettük a magyar beszéd fonetikai szabályait a szövegkörnyezet függvényében. Ilyenek például az alkalmazkodási folyamatok (hasonulás, asszimiláció, stb.).
A folyamatos beszédre tipikusan jellemző, hogy a szavak között nincs szünet. A fizikai paraméterek folyamatosan változnak a szóhatárokon. Ezért a további feldolgozás megkönnyítése érdekében bejelöltük a szavak határait is.

2.4 Kézi szegmentálás

Az adatbázis annotálása után következő feladat annak fonetikai szintű szegmentálása és címkézése, valamint a szóhatárok és a frázishatárok bejelölése. A feladat „audiovizuális fonetikai átírás" a 1997-ban elkészült BABEL nemzetközi project leírása

alapján (Vicsi K., Vig A.). Az átírást a szöveg hallgatása, és az időfüggvény és/vagy a színkép elemzése alapján hajtottuk végre.
Az aktuális kimondásnál kimaradt hangokat megjelöltük, a követő hang előtt zárójelbe tettük.
A **szószintű szegmentálás** esetén a szavak határát jelöltük. A **mondatok határait** a frázisjelző írásjelek (vessző, kettőspont, pontosvessző, gondolatjel, macskaköröm, és, pont kérdőjel, felkiáltójel) adták.

3. A Magyar Referencia Beszédadatbázis adatkészleteinek átfogó műszaki tulajdonságai

Magyar nyelvű, olvasott szövegű, személyi számítógépes környezetben felvett adatbázis, 16 bites, 16 kHz-es mintavételezéssel.

- *332 beszélő közvetlenül a számítógépbe rögzített hanganyaga;*
- *Beszélőnként 12 mondat és 12 szó*
- *A felvételek többféle mikrofonnal, hangkártyával, PC-kel készültek*
- *Környezet változó zajosságú irodahelyiség, laboratórium, otthoni környezet;*
- *Az adatbázis teljes anyaga annotált, az adatbázis harmada (100 beszélő) kézileg szegmentált és címkézett.*

4. Köszönetnyilvánítás

A kutatás az OTKA T 046487 ELE és az IKTA 00056 pályázatok keretén belül készül.

5. Irodalom

Vicsi, K. Beszédadatbázisok a gépi beszédfelismerés segítésére, Híradástechnika, Vol. 2001/1, Budapest, pp. 5-13, 2001.

Vicsi, K. Vig, A.: Az első magyar nyelvű beszédadatbázis, Beszédkutatás'98, Tanulmányok az elméleti és alkalmazott fonetika köréből. MTA Nyelvtudományi Intézet, Budapest, pp. 163-178, 1998.

Wells, J. at all.: Standard Computer-Compatible Transcription. Esprit Project 2589 (SAM), Doc. no. SAM-UCL-037. London: Phonetics and Linguistics Dept., UCL (1992).

Fourcin, A.J. and Dolmazon, J-M. „Speech knowledge, standards and assessment", Proceedings of XII International Congress of Phonetic Sciences, Aix-en-Provence, Vol. 5, 430-433 (1991).

Folyamatos beszéd szó- és frázisszintű automatikus szegmentálása szupraszegmentális jegyek alapján

Vicsi Klára, Szaszák György, Borostyán Gábor

Budapesti Műszaki és Gazdaságtudományi Egyetem, Távközlési és Médiainformatikai Tanszék, Beszédakusztikai kutatólaboratórium
{vicsi, szaszak}@tmit.bme.hu
http://alpha.tmit.bme.hu/speech/

Abstract. Cikkünkben a beszéd alapfrekvencia- és energiaviszonyainak vizsgálatával arra keressük a választ, lehetséges-e ezen prozódiai beszédjellemzők alapján valamilyen módon a folyamatos beszéd gépi tagolása frázisok, illetve szószerkezetek, szavak szintjén. Mindezzel a folyamatos gépi beszédfelismerő működését segíthetnénk a szavak, szószerkezetek határainak detektálásával, ezáltal jelentősen lecsökkentve a beszédfelismeréskor a dekódolás során a keresési teret. Kitérünk az egyes algoritmusokkal elért eredmények bemutatására is. A vizsgálatokat statisztikai módszerekkel végeztük az olvasott szöveget tartalmazó BABEL beszédadatbázison. Várhatóan spontán beszédet tartalmazó szövegben a döntési biztonság az itt bemutatandóhoz képest csökken.

1 Bevezetés

A beszédet az artikulációs szervek folyamatos mozgatásával hozzuk létre. A produktum, vagyis a levegőben terjedő nyomáshullámok eszerint a folyamatos mozgás szerint alakulnak. Vizsgálva a keltett nyomáshullámok fizikai paramétereit, azt tapasztaljuk, hogy ezek a paraméterek is folyamatosan változnak, például a szavak között nem tartunk szünetet. Ha mindig szünetet tartanánk, beszédünk akadozóvá válna. A beszédben szavak, szószerkezetek határait csak egy nyelv megtanulása után vagyunk képesek észlelni, magasabb szintű agyműködés eredményeként. A prozódiai jegyek, az alapfrekvencia, az intonáció és az időtartamarányok segítik a beszéd tagolását. Míg a mondat modalitásának kialakulásában az alapfrekvencia menetének egyértelműen meghatározó szerepe van, addig a hangsúly esetében már nem ilyen egyértelmű a helyzet. A szubjektív hangsúlyérzet kialakulásában mindhárom jellemző részt vesz, egymással szoros összefüggésben. Például az intenzitás emelkedése fiziológiai okokból maga után vonja az alapfrekvencia növekedését is, mivel a megemelkedett szubglottális nyomás a hangszalagokat egyúttal szaporább rezgésre kényszeríti [1]. Legnagyobb eséllyel az a szótag kelt a hallgatóban hangsúlyélményt, melynek mind alapfrekvenciája, mint intezitása kiemelkedő. E kettő közül is alapvetőbb az alapfrekvencia, mert keletkezhet hangsúlyélmény kiemelkedő alapfrekvencia esetén akkor is, ha az intenzitás a környező szótagok intezitásánál valamivel kisebb.

A magyar nyelv kötött hangsúlyú nyelv, a hangsúly az első szótagon szokott lenni. A kötött hangsúlyú nyelvekben a hangsúly mondatszinten értelmezhető jól, a szószerkezetek, illetve a mondat hangsúlyosságát a beszélő szándéka, valamint a mondatszerkesztés szabályai határozzák meg. Ennek megfelelően szakaszhangsúlyról és mondathangsúlyról beszélhetünk.
Korántsem minden szó hangsúlyos tehát, az azonban biztos, hogy ha a mondat valamely szótagja hangsúlyt kap, ez a szótag szinte biztosan szó elején található [3]. Folyamatos gépi beszédfelismerésnél éppen ezért támpontot adhat az, ha a hangsúlyt detektálni tudjuk, mert tagolni tudjuk a beérkezett fonémafolyamot, valamint abban fix pontokat határozhatunk meg, mely által a felismerés hatékonysága is javul. Vélelmezhetjük továbbá, hogy a hangsúlyos pozícióban lévő fonémák felismerése is biztosabb volt (a hangsúlyozás igen sokszor párosul gondosabb artikulációval [2]).
Nem szóltunk még egy fontos prozódiai elemről, a beszédszünetekről. Szünet alatt a prozódiában nem csak a tényleges akusztikai jelkimaradás értendő, hanem másodlagos szünethordózókat is ismerünk. Ezek lehetnek hangzónyújtás, glottális zár, hirtelen hangmagasság-változás, hasonulások elmaradása, illetve ezek kombinációi [2]. Rejtett Markov modelles felismeréskor az akusztikai jelkimaradásra, mely egyben elsődleges szünethordozó, külön modell készíthető, ennek gépi detektálása tehát megoldható. Egyes másodlagos szünethordozók pedig vizsgálhatók a hangsúly detektálásra használt algoritmusokkal. A szünet detektálásának jelentősége ugyan az, mint a hangsúly esetében: utána biztosan új szó kezdődik, ezáltal a felismeréskor kapott fonémafolyam tagolható.

2 Vizsgálati alapelvek

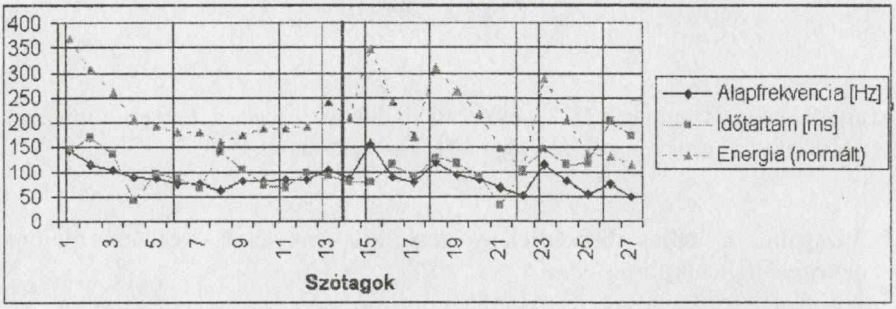
A hangsúly automatikus meghatározásához fizikailag jól mérhető paraméterekre van szükség, előzetes vizsgálataink során az alapfrekvencia és az intenzitásszint értékeit találtuk megbízhatónak. Az időtartam felhasználása két szempontból is problematikus lenne: egyrészt a beszédfelismerés során igen pontosan kellene ismernünk az egyes hangok helyét ahhoz, hogy pontos mérőszámot adhassunk a szótagok hosszára, másrészt a szegmentált anyagok átnézésekor azt tapasztaltuk, hogy a tényleges hangsúllyal a szótagok hossza csak kis mértékben korrelált. A hangsúly végső detektálásához tehát az alapfrekvencia és az energiaszint értékeit figyelembe vesszük (1. ábra).
A hangsúly detektálására kétféle algoritmust dolgoztunk ki. Az első módszernél a hangsúly detektálását csak a szótagok magánhangzóinak kvázistacioner részén mért alapfrekvencia (Hz) és energiaszint (dB) értékeket használjuk. A második módszernél a teljes hanganyagon mért alapfrekvencia és energiaszint értékek alapján történt a hangsúly detektálás. A kétféle algoritmust az alábbiakban mutatjuk be (lásd 2.1. és 2.2. pontokat).

Vizsgálatainkat a BABEL magyar nyelvű, olvasott szöveget tartalmazó adatbázison [4] végeztük, férfi beszélőkre. Az adatbázis fonéma szinten szegmentált, illetve járulékosan szupraszegmentális információkat is tartlmaz. Ezek a szó-, frázis- és mondathatárok helyeinek, valamint a mondattípusok külön jelölését jelentik. A teljes vizsgálat mintegy 1600 mondatnyi anyagon zajlott le, mely 22 beszélőtől származott.

6. ábra: a hangsúly detektálás elvi vázlata

2.1 Hangsúly detektálás szótagok magánhangzóin mért paraméterek alapján

Ebben az esetben a hangsúly vizsgálatakor a szótag magánhangzójának kvázistacioner részét vettük alapul, ezen mértük az alapfrekvencia és az energiaszint értékeit.



7. ábra Alapfrekvencia, energiaszint és időtartam viszonyok a 'Gróf Vásárhelyi Görögországban kötött ki, és titkárul szerződtette a főkonzul lányát.' mondat szótagjainak magánhangzóinak kvázistacioner részein mérve. Az x tengelyen a szótagok sorszáma látható.

A 2. ábrán láthatjuk a 'Gróf Vásárhelyi Görögországban kötött ki, és titkárul szerződtette a főkonzul lányát.' magyar mondat esetén a szótagok magánhanzóinak stacioner részén átlagolt alapfrekvencia és energiaszint-értékeket, valamint a szótag magánhangzóinak hosszát. (Az energiaszint-értékek az egy ábrán való megjeleníthetőség kedvéért lineárisan áttranszformáltak.) A hangsúly detektálásához csúcskeresési algoritmusokat használtunk. Ezek lényege, hogy kiszámítjuk az adott x_i adatsor várható értékét és szórását, majd ezekből egy

$$K = M + k * \sigma \qquad (1)$$

küszöböt határozunk meg, ahol k tetszőleges konstans általában 0.5 – 1.5 közötti értékkel. Ezt követően minden x_i -re megvizsgáljuk, nagyobb-e a K küszöbnél, ha igen, akkor ezt csúcsnak tekintjük, és itt hangsúlyos pozíciót detektálunk.
Magyar nyelvű kijelentő mondatot alapul véve mind az alapfrekvencia-, mind az energiaszint folyamatos csökkenést mutat. Ennek kompenzálására a küszöböt csúszóablakkal számítjuk, az ablak méretét 7 – 17 szótag között célszerű választa-

nunk. Ezáltal a küszöböt a mondat dallammenetéhez igazítjuk. Az *i*-edik szótaghoz tartozó küszöb tehát:

$$K_i = M(x_{i-A}, x_{i-A-1}, ..., x_i) + k * \sigma(x_{i-A}, x_{i-A-1}, ..., x_i) \text{, ha i > A} \quad (2)$$

$$K_i = M(x_1, x_2, ..., x_A) + k * \sigma(x_1, x_2, ..., x_A) \text{ egyébként.} \quad (3)$$

ahol *A* a csúszóablak mérete szótagszámban kifejezve.
Hasonlóan, az egyes szótagok közötti alapfrekvencia- és energia differenciáit is számítottuk, melyeken ugyanezt a csúcskeresést futattuk le azzal a különbséggel, hogy a várható érték (4) és a szórás (5) számításakor a kapott értékek abszolút értékeit vettük:

$$M_i = \frac{1}{A} \sum_{j=i-A}^{i} |\Delta x_j| \quad (4)$$

$$\sigma_i^2 = \frac{1}{A} \sum_{j=i-A}^{i} \left(M_j - |\Delta x_j|\right)^2 \quad (5)$$

A csúszóablakos számítás (2),(3) ekkor is indokolt, mivel a levegő fogytával a beszéddinamika is csökken a frázis vége felé, kijelentő mondatban.

2.2 Vizsgálat a teljes beszédjelen mért alapfrekvencia- és energiamenet jelleggörbéje alapján

Felmerült, hogy a hangsúlydetektáló algoritmust ne csak a szótagok magánhangzóin mért értékek alapján, hanem a teljes hanganyag folytonosnak tekintett alapfrekvencia- és energiaszint-menete alapján próbáljuk megalkotni. Az E_i energiagörbét nagy, 100 ms-os integrálási idővel számítjuk, hogy a gyors, kismértékű fluktuációt kiszűrjük. Ezután ismét átlagoljuk a görbét M = 125 ms-os csúszóablakkal, így kapjuk meg az E_i' görbét (6), majd az eredeti E_i energiagörbe e fölé eső részeit tartjuk csak meg, ebből adódik E_i'' (7).

$$E_i' = \frac{1}{M} \sum_{m=i-\frac{M}{2}}^{i+\frac{M}{2}} E_m, \quad M = 125\text{ms} \quad (6)$$

$$E_i'' = E_i', \text{ ha } E_i \geq E_i' \quad (7)$$

$$E_i'' = 0 \text{ egyébként}$$

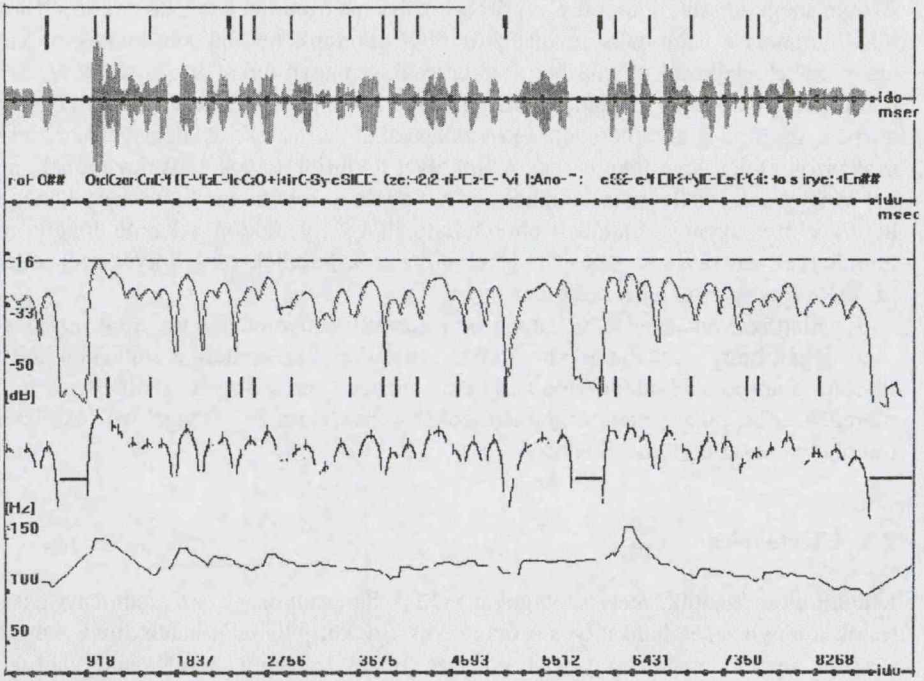
Ezután megkeressük a kapott E'' görbe lokális maximumhelyeit, de ekkor két lokális maximumhelyre minimális távolságküszöböt iktatunk be: ha két lokális maximum ennél közelebb kerül egymáshoz, akkor csak a nagyobbikat fogadjuk el. A lokális maximumhelyek által meghatározott pontokra görbét illesztünk. (3. ábra) A burkológörbén végül negatív meredekségű szakaszokat keresünk. A negatív meredekségű szakaszok elején lévő lokális maximumhelyet tekintjük hangsúlyos pozíciónak, a szóhatárt így ezen lokális maximumhely és a megelőző lokális maximumhely közé jelezhetjük előre, ugyanis általában elmondható, hogy ily módon a kapott lokális maximumhelyek az egyes szótagok magánhangzóinak felelnek meg, mivel ezek energiaszintje a legnagyobb a beszédjelben.

Az alapfrekvencia görbén lényegében hasonló műveleteket végzünk, azzal a különbséggel, hogy a lokális maximumokat közvetlenül az eredeti, a zöngétlen helyeken lineáris interpolációval folytonossá tett görbén keressük. A szóhatár a negatív meredekségű szakasz elején található lokális maximum és a megelőző lokális minimumhely között kerül detektálásra.

2.3 Kiértékelés

Miután előrejeleztük, mely szótagokat vélünk hangsúlyosnak, az eredményt összevetettük a ténylegesen hangsúlyos szótagokkal, így két jellemzőt határoztunk meg. Egyrészt az algoritmus hatékonyságát, hogy az összes szó hány százalékánál találtuk meg a szóhatárt, másrészt, a pontosságot, hogy milyen pontos volt az előrejelzés, azaz volt-e olyan hangsúlytalan szótag, melyet hangsúlyosnak osztályozott a program. A felhasználás szempontjából ez utóbbi a különösen kritikus érték, ugyanis ha felismerés során erre az osztályozásra szeretnénk támaszkodni, nagy pontosságot kell elérni. Sajnos a BABEL adatbázis nem tartalmazza a hangsúlyok címkézését, így minden szó eleji szótagot potenciális hangsúlyos pozíciónak tekintettünk, a hatékonyság emiatt nehezen értelmezhető mutató, hiszen a valós beszédben sem hangsúlyos minden szóindító szótag. Mindazonáltal az egyes módszerek összehasonlítására alkalmas.

A jelleggörbék alapján végzett detektálás esetében fontos látnunk a különbséget az előző módszerhez képest abban a tekintetben, hogy míg az előbb szótagokon mérve vizsgáltuk azok hangsúlyosságát, addig most a teljes jelfolyamon dolgozunk, melyről azután a hangsúlyos pozíciót szótag szintre vissza kell képezni. Akkor tekintettük szóhatár-predikciónkat sikeresnek, ha annak 100 ms-os környezetébe esett a tényleges szóhatár.



8. ábra szóhatár detektálás illusztrációja: a teljes beszédfolyamon mért időfüggvény (fönt) energia és lokális maximum-helyek (középen) és az alapfrekvencia (lent) görbék alapján.

3 Eredmények

A 2.1. pontban bemutatott csúcskereső algoritmusok használatakor különböző *k* konstans és *A* szótagszámban mért ablakszélesség értékek mellett vizsgáltuk a hangsúlyok detektálásának pontosságát és hatékonyságát.

Hatféle kiértékelést készítettünk, ezek az alapfrekvencia, az energiaszint, valamint az alapfrekvencia- és energiamenet alapján együttesen hangsúlyosnak osztályozott pozíciók, valamint az alapfrekvencia-változás, az energiaszint-változás, illetve ezek együttes megléte esetén hangsúlyosnak vélt pozíciók. Az eredmények az 1. táblázatban láthatóak.

Látható, hogy nagyobb, 10 szótag fölötti csúszóablak-szélesség beállítással a pontosság valamelyest növelhető, ez általában maga után vonja a hatékonyság kismértékű csökkenését is. A *k* konstans értékének növelésével – ahogyan az várható – egyértelműen növekszik a pontosság, de a hatékonyság nagyobb mértékben esik az ablakszélesség növelésekor tapasztaltnál.

9. táblázat. Hangsúly detektálás pontossága és hatékonysága magánhangzók kvázistacioner részén mért paramétereinek alapján

A	k	Pontosság/Hatékonyság [% / %]					
		F_0	E	F_0 & E	ΔF_0	ΔE	ΔF_0 & ΔE
7	0.5	49/44	46/30	46/20	76/24	57/21	82/10
7	0.7	50/39	45/27	46/16	77/23	58/19	83/10
7	0.9	51/33	45/24	47/13	78/21	59/17	86/9
7	1.1	52/28	45/21	47/10	79/20	60/15	87/7
9	0.5	49/41	46/29	45/18	76/24	59/21	84/11
9	0.7	50/36	46/26	47/15	77/22	60/19	83/9
9	0.9	52/32	46/23	47/12	78/21	61/17	83/9
9	1.1	52/27	45/20	47/9	79/19	62/15	85/8
13	0.5	51/39	45/27	46/16	77/22	61/19	84/9
13	0.7	52/34	45/23	46/13	78/20	63/18	84/8
13	0.9	52/28	45/20	46/11	79/19	64/16	87/8
13	1.1	54/24	46/18	49/9	79/17	65/14	88/7
17	0.5	51/38	46/26	46/16	78/21	64/19	86/9
17	0.7	53/33	45/22	47/10	78/19	63/17	86/8
17	0.9	54/28	46/20	49/10	79/18	65/15	86/7
17	1.3	56/20	46/15	52/7	81/15	65/11	90/6

Az eredmények mind az energiaszint, mind az energiaszint-változás esetén jóval gyengébbek az alapfrekvenciával kapottaknál. Ennek részben oka lehet, hogy a magánhangzók energiájának szubjektív észlelése függhet a magánhangzótól. A nyitottabb ajkakkal képzett magánhangzók nagyobb energiájúak, emiatt előfordulhat, hogy hangsúlyos, de kerekítettebb ajkakkal képzett magánhangzó energiája kisebb a hangsúlytalan nyílt magánhangzóénál.

A teljes beszédjel alapfrekvencia-, illetve energiamenet jelleggörbe alapján (lásd 2.2. pont) kapott eredményeket a 2. táblázatban láthatjuk. Az eredményeket az 1. táblázat ΔF_0, ΔE, illetve ΔF_0 & ΔE oszlopokban kapott eredményeivel érdemes összevetnünk, mivel a lokális maximumhelyek megkeresése, illetve a jellegörbék tulajdonságainak vizsgálata is delta dimenziójú paramétereken nyugszik. Mindezek alapján elmondhatjuk, hogy az alapfrekvencia alapján a predikció pontatlanabb, igaz valamivel hatékonyabb is. Az energiaszint esetében egyértelműen a második megközelítéssel kapunk jobb eredményt. Ennek oka a fentiekhez hasonlóan valószínűleg az, hogy a zárt magánhangzókat relatíve hangsúlyosabbnak érzékeljük kisebb energiatartalom esetén is, ezzel a módszerrel azonban jobban megfogható ez a jelenség. Az együttes becslés pontosságában és hatékonyság tekintetében szintén felülmúlja az egyszerű csúcskereséssel kapott értékeket. Az összehasonlításkor azonban legyünk óvatosak, mert a két módszer közül a második megítélésekor a mérési pontatlanság nagyobb.

10. táblázat. Hangsúlyos pozíció detektálása a teljes beszédjelen mért jelleggörbék alapján

Pontosság/Hatékonyság [% / %]		
F_0	E	F_0 & E
70/32	69/34	91/14

Összességében elmondhatjuk, hogy bármely módszer felhasználhatóságához megközelítőleg legalább 85%-os, de lehetőleg minél nagyobb pontosság elérése a kívánatos. Ezt az értéket jelen vizsgálatunkban sikerült elérni, 9-14% közötti hatékonyságot kaptunk ebben az esetben a különböző mérési elrendezésekben. Megjegyzendő, hogy a beszédfolyamat során sem minden szó hangsúlyos, ezért a hatékonyságban véleményünk szerint 50%-ot meghaladó eredményt elérni nem is lehetne.

4 Konklúzió

Cikkünkben a folyamatos beszéd automatikus szegmentálásának problematikáját tárgyaltuk. Bemutattuk ennek szerepét a beszédfelismerés során, valamint a hangsúly detektálására kidolgozott két módszert. A kapott eredmények alapján úgy gondoljuk, érdemes a területen tovább vizsgálódni, a közeljövőben tervezzük az alapfrekvencia és energiaszint paraméterek alapján a hangsúly előrejelezhetőségének vizsgálatát statisztikai módszerekkel, valamint a hangsúlydetektáló modult beszédfelismerőbe építve vizsgálni szeretnénk, javítható-e és milyen mértékben a beszédfelismerés hatékonysága.

Köszönetnyilvánítás
A kutatás az OTKA T 046487 ELE és az IKTA 00056 pályázatok keretén belül készül.

Irodalomjegyzék

1. Kassai Ilona: Fonetika. Nemzeti Tankönyvkiadó, Budapest (1998)
2. Kassai Ilona – Fagyal Zsuzsanna: Hogyan észlelik a magyar beszéd szüneteit magyar és francia anyanyelvű hallgatók. In: Magyar nyelvőr, Budapest (1996/120) 209-220. o.
3. Kiefer Ferenc (szerk): Struktúrális magyar nyelvtan, II. kötet, Fonológia. Akadémiai Kiadó, Budapest (2000).
4. Vicsi Klára – Vig Attila: Az első magyar nyelvű beszédadatbázis, Beszédkutatás'98, Tanulmányok az elméleti és alkalmazott fonetika köréből. MTA Nyelvtudományi Intézet, Budapest (1998) 163-178. o.

Az automata és kézi szegmentálás ejtésvariációk okozta problémái

Zsigri Gyula[1], Tóth László[2], Kocsor András[2], Sejtes Györgyi[1]

[1] Szegedi Tudományegyetem Magyar Nyelvészeti Tanszék
{zsigri, sejtes}@hung.u-szeged.hu
[2] MTA-SZTE Mesterséges Intelligencia Kutatócsoport
{tothl, kocsor}@inf.u-szeged.hu

Kivonat: A beszédadatbázisok egyik legértékesebb része a beszédhang szintű szegmentálási információ. A szegmentálást és címkézést tökéletesen csakis nagy figyelmet igénylő, fáradságos és hosszadalmas kézi munkával lehet elvégezni. Megkönnyítheti és meggyorsíthatja viszont a munkát egy speciálisan erre a célra kialakított algoritmus, amely megkísérli automatikusan elhelyezni a fonetikai határokat. Akár ember, akár gép végzi a szegmentálást, segítségként rendelkezésére áll a hanganyag feltételezett fonetikai átirata, amelyet egy fonetikus átíró algoritmus állít elő a betű szerinti lejegyzésből. A jel valódi fonetikai tartalma azonban eltérhet ettől, hiszen ugyanannak a szövegnek ejtésvariációja lehet. A cikkben megvizsgáljuk, hogy ez a jelenség hogyan befolyásolja az általunk alkalmazott automata, illetve félautomata szegmentáló algoritmusokat. Megnézzük továbbá, hogy az MTBA adatbázis kézi feldolgozása során a szegmentálást végző személyek miben tértek el az előzetesen rögzített szabályoktól, különös tekintettel arra, hogy mentális (fonetikai) lexikonjuk hogyan befolyásolta őket a várttól eltérő ejtésvariációk kezelésében.

1 A szegmentálás

A statisztikai alapú beszédfeldolgozáshoz, különösen a gépi beszédfelismeréshez jól megszerkesztett, nagyméretű beszédadatbázisokra van szükség. A felismerők betanítása nem más, mint statisztikai alapú paraméterbecslés. A pontos becsléshez nagyszámú minta alapján történő betanítás szükséges. E minták gyűjteményei – a szükséges jegyzetekkel, címkézésekkel és átírásokkal – képezik az adatbázist. Az adatbázisoknak lehetőleg tartalmazniuk kell mindazokat a mintákat, amelyek egységesen lefedik a beszéd (és a környezeti zajok) változatosságát.

Az adatbázis legfontosabb része a szegmentált hanganyag. A szegmentálás a beszédfolyamat lineáris tagolása, vagyis a hangtest hangegységekre bontása. A hangtestben diszkrét metszetek kijelölésével jutunk el a beszédhanghoz, a hangtest legkisebb szegmentális szerkezeti egységéhez [1]. A beszéd időfüggvényében bejelöljük a fizikailag megfigyelhető beszédhangokat és azok határait. A szegmentálás célja, hogy a gépi feldolgozáshoz megadjuk a beszédjel és a fonetikai átirat közti kapcsolatot, azt, hogy melyik szimbólum a beszédjel mely időintervallumának felel meg. A szegmentálás egységei a beszédhangok, ezekből absztraháljuk a fonémákat [6].

A fonetikai szintű szegmentálás és címkézés általában kézileg történik; az átírásban a szöveg lehallgatása, továbbá az időfüggvény és/vagy a színkép elemzése nyújt segítséget. Ebben az ún. „audiovizuális fonetikai átírásban" az 1997-ben elkészült BABEL nemzetközi project ajánlásait [7] követtük minden ilyen jellegű munkánkban. A feldolgozás elvégzéséhez kifejlesztettünk egy speciális célprogramot, amely képes megjeleníteni a beszédjel hullámképét, illetve a színképelemzés révén előálló ún. spektrogramot. A lehallgatás mellett ez a két vizuális információ segíti a szegmentálást. A beolvasott szöveg alapján a program előállít, és felkínál továbbá egy feltételezett fonetikus átiratot is; ez a fonetikus átirat természetesen javítandó, ha a beszélő esetleg mást mondott, mint amit a szoftver feltételezett. Szegmentálóprogramunk az SZT-IS-10 pályázat keretében készült, és ingyenesen letölthető a http://www.inf.u-szeged.hu/oasis címről, illetve további részleteket is itt találhatunk róla.

2 Automatikus szegmentálás

2.1 Automatikus szegmentálás kényszerített illesztéssel

A beszédhang szintű szegmentálást és címkézést csakis nagy figyelmet igénylő és hosszadalmas kézi munkával lehet elvégezni. Megkönnyítheti és meggyorsíthatja viszont a munkát egy megfelelő, speciálisan erre a célra kialakított algoritmus, amely megkísérli automatikusan elhelyezni a fonetikai határokat. Bár ezt a feladatot jelenlegi tudásunk szerint tökéletesen nem tudjuk megoldani, olyan program azért készíthető, amely a határok jó részét elég pontosan helyezi el. A szegmentáló személy feladata ilyenkor csak az automata szegmentáló javaslatainak ellenőrzése és korrekciója, ami által a szegmentálás felgyorsítható.

Az automata szegmentálási algoritmusok közül a legjobbak azok, amelyek gépi tanuláson alapulnak. Speciálisan, a gépi beszédfelismerők is felhasználhatók a szegmentálási feladat elvégzésére. A beszédfelismerő algoritmusok ugyanis egy mondat felismerése közben keresést végeznek: a felismerendő hangjelre megpróbálják ráilleszteni az összes lehetséges fonetikai átiratot. Mivel a felismerés során a beszédhangok határai sem ismertek, ezért a felismerők végigpróbálják az összes lehetséges szegmentálást is. A felismerés eredményeképpen azt az átiratot, illetve szegmentumhatár-sorozatot adják vissza, amelyet az adott jel esetén a legvalószínűbbnek találtak. Egy beszédfelismerési alkalmazás esetén persze nincs szükségünk a szegmentumhatárokra, így az eredménynek ezt a részét figyelmen kívül hagyjuk. Viszont a felismerőnek ez az amúgy rejtve maradó „képessége" remekül kihasználható az automatikus szegmentálás céljaira. Ilyenkor ráadásul könnyebb a feladat, mint a felismerés esetén, ugyanis a fonetikai átirat adott, így azt nem is kell keresni; csupán az adott átirat és a jel közötti optimális beszédhanghatár-összerendelést kell megtalálni. A beszédfelismerők ilyesfajta felhasználását „forced alignment"-nek nevezi a szakirodalom.

Eddigi munkáink alapján mi is készítettünk egy automata szegmentáló rutint a fent leírt módon beszédfelismerő felhasználásával. A beszédfelismerő az MTA-SZTE Mesterséges Intelligencia Kutatócsoportnál fejlesztett „OASIS" rendszer volt, amely beszédhang alapú, és mesterséges neuronhálókat alkalmaz a beszédhang-felismerésre. A módszer technikai hátterét korábban már részletesen ismertettük [9], így attól itt

most eltekintünk. A továbbiakban inkább arra térünk ki, hogy az automatikus szegmentáló használata során milyen nehézségek jelentkezhetnek, és ezeken hogyan lehet segíteni.

2.2 A kényszerített illesztés problémái

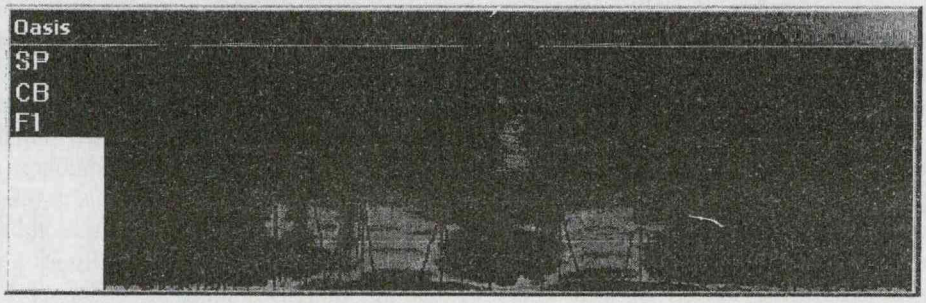
Habár jelenleg a beszédfelismerők „kényszerített illesztéses" felhasználása tűnik a legjobb automatikus szegmentálási módszernek, ez sem mentes a maga problémáitól. Ezek közül a legfontosabb, hogy az illesztés során használt fonetikus átirat esetenként lényegesen eltérhet a szegmentálandó beszédjel valódi tartalmától. A fonetikus átiratot ugyanis algoritmikusan állítjuk elő a beolvasott szöveg betű szerinti leírásából. Egy adott mondatnak pedig többféle eltérő kiejtése is lehet. Az alternatívák főleg a szóhatárok hasonulásainak végbemeneteléből vagy hiányából adódnak, de más tényezők is befolyásoló hatásúak, mint például a beszédsebesség vagy a beszélő artikulációs igényessége (vagy igénytelensége). Az ejtésvariációk kezelése nemcsak az automatikus szegmentálás esetén, hanem a beszédfelismerők szótárának automatikus kialakításánál is kulcsfontosságú, ezért vizsgálatuk máris megindult [8]. A Műegyetemen fejlesztett fonetikus átírórendszer kiejtési opciókat is megenged a fonetikai átiratban, és az eredmények arra utalnak, hogy ez valóban valamivel jobb felismerési eredményekhez vezet [2]. Azonban az MTBA adatbázis feldolgozása során nekünk nem állt rendelkezésünkre ilyen képességű fonetikai átíró, így az illesztések során csak egyfajta fonetikai átirat szolgált inputként. Ezzel gyakran előfordult, hogy bizonyos pontokon – például szóhatároknál – két-három egymást követő fonetikai szimbólumban sem felelt meg a beszédjelnek. Ilyenkor a beszédfelismerő rendszer kénytelen ráerőltetni a hibás szimbólumokat az adott jelszakaszra, illetve túl nagy eltérések esetén az algoritmus nem is hajlandó lefutni. Ha lefut is, ezeken a pontokon nyilván hibás eredménnyel. Ez pedig azt a veszélyt hordozza magában, hogy a későbbi manuális korrekció során a feldolgozó személy hajlamos átsiklani ezeken a részeken, és elfogadja a gép által kínált hibás megoldást.

2.3 Automatikus szegmentálás határtanulással

A beszédjelek automatikus úton történő beszédhangokra szegmentálása egyike a számítógépes beszédfeldolgozás klasszikus problémáinak. A tökéletes megoldásnak (amennyiben egyáltalán lehetséges) mindenképpen valamiféle tanulási módszeren kell alapulnia. Olyan technikák, amelyek pusztán jelfeldolgozáson alapulnak és az éppen adott szegmentálandó jelen kívül mást nem ismernek, nem tudnak teljes értékű megoldást szolgáltatni (ezt a megoldást ezért „félautomatának" neveztük, arra utalva, hogy a szegmentáló kimenetét utólag kézzel még korrigálni kell). Ezek az egyszerű módszerek a jel változását mérik, és a nagy változásoknál jeleznek feltételezett szegmentumhatárt. Korábbi munkánkban mi is kidolgoztunk egy olyan jelfeldolgozó függvényt, amely a szegmentumhatárokat hivatott jelezni [9]. Ez a függvény a spektrum megfelelően kiválasztott sávjainak energiáit, pontosabban azok változását vizsgálja. Az egyes sávokra illesztett detektáló függvényeket súlyozott összegzéssel kombináltuk, ahol a megfelelő súlyértékeket tapasztalati úton, hosszas kísérletezéssel lőttük be. Ez természetesen nem volt optimális, ráadásul az egyik adatbázison jónak talált

értékek nem feltétlenül működtek egy másikon. Ezen javítandó megkíséreltük a paraméterek optimális értékét gépi tanulással megtalálni. Ez a következőképpen történt:

Tanító adatbázisnak az adatbázis kézileg már felszegmentált részét használtuk. Elvileg a határként bejelölt időpontok beszédkereteit kellett volna pozitív tanulópéldaként használnunk, a határok közé eső kereteket pedig negatív példaként. Ezzel az egyik probléma az, hogy ily módon jóval több lett volna a negatív példa, mint a pozitív, korábbi tapasztalataink szerint pedig ez gondot okozhat a tanulásnál. A másik nehézség pedig, hogy maguk a kézileg behúzott határok is lehetnek némileg pontatlanok. Ezért azt a megoldást választottuk, hogy a határok közé egy x^6 jellegű görbét illesztettünk oly módon, hogy az a két határnál vegye fel az 1-es értéket, a szegmentum középpontjánál pedig a 0-át. Ily módon a „határság" valószínűségét kissé „szétkentük", a kézileg behúzott határoktól való távolság függvényében. Az így előállt célfüggvény tanulására (regressziójára) egy kétrétegű előrecsatolt neuronhálót alkalmaztunk 20 rejtett neuronnal. Input jellemzőkészletként a kézi szegmentáló kifejlesztésekor kikísérletezett, sávenergiákból származtatott görbék szolgáltak. A tanulás után a neuronháló által szolgáltatott görbét szemlélteti a 1. ábra.



1. ábra: A *'kilencven'* szó spektrogramja, a neuronháló által szolgáltatott görbe, és az ez alapján algoritmikusan kiválasztott szegmenshatárok

A neuronháló kimenetéből legegyszerűbben úgy kaphatunk szegmentumhatárokat, ha megkeressük a lokális maximumhelyeit. Továbbá a görbe adott pontban felvett értéke valószínűségi értékként értelmezhető, így egy egyszerű küszöböléssel szabályozhatjuk a behúzott határok számát, ekképp egyensúlyozva a törlési és beszúrási hibák között. A 1. ábrán bejelölt határokat is a lokális maximumhelyekből kiindulva állítottuk elő. Tapasztalataink szerint a neuronhálós megoldás sokkal stabilabban működik, mint a korábbi, empirikus úton belőtt félautomata szegmentálónk, és bizonyos fajta hanghatárokat nagyon jó pontossággal talál el. Hibákat főleg olyan esetekben követ el, amely esetek a kézi szegmentálás során is a legnehezebbeknek bizonyulnak (pl. magánhangzó-félmagánhangzó átmenet). Ha a pontos fonetikus átirat adott, akkor természetesen ez a fajta szegmentálás rosszabb eredményt ad, mint a kényszerített illesztés, hiszen a fonetikus átirat ismerete nélkül kell dolgoznia. De ha a fonetikus átirat komoly hibákat tartalmaz, akár jobbnak is bizonyulhat. Továbbá a küszöbérték beállításával elérhető, hogy az algoritmus csak a biztos helyekre húzzon, a több határt hagyja ki. Ilyenkor a kézi ellenőrzést és korrekciót végző személy kénytelen elhelyezni a további hiányzó határokat, míg a kényszerített illesztéssel dolgozó algoritmus mindig a szükséges számú határt húzza be, amelynek hibáin a figyelmetlen szemlélő esetleg átsiklik.

2.4 Határtanulás a beszédfelismerésben

Az előző fejezetben ismertetett határtanulási módszer a beszédfelismerésben is hasznos lehet. Mint korábban említettük, futásuk során a felismerők gyakorlatilag végigvizsgálják az összes lehetséges fonetikai szegmentálást. Ez egy óriási hipotézistér bejárását jelenti, nyilván komoly időigénnyel. Amennyiben bizonyos szegmentálási lehetőségeket egy gyors algoritmussal ki tudunk zárni, akkor ezeket az eseteket nem kell a felismerőnek kiértékelnie, így gyorsabb működést érhetünk el. Természetesen mindig fennáll a kockázata annak, hogy a fürdővízzel a gyereket is kiöntjük, azaz véletlenül egy valódi határt is kidobunk. Ezt a felismerő többnyire már nem képes korrigálni, ezért inkább túl sok, mint túl kevés határt kell behúznunk. Az előző fejezetben ismertetett algoritmus használata esetén egyszerűen a küszöbérték módosításával szabályozhatjuk, hogy hány határ maradjon meg.

Ha számszerűsíteni akarjuk egy automatikus szegmentáló hatásfokát, akkor legjobb, ha összevetjük a kézi szegmentálással. Erre a következő, "edit distance" jellegű algoritmust alkalmaztuk: a gépi és a kézi határsorozat határait összepárosítjuk úgy, hogy a párok távolságainak összértéke minimális legyen (ez dinamikus programozással megoldható). Ha a két határsorozat nem ugyanannyi határból áll, akkor nyilván lesznek kimaradó határok. Nem fogadjuk el tovább azokat a párokat, amelyeknél a távolság egy ezredmásodpercben adott küszöbnél nagyobb. Végeredményként a kézi szegmentálás pár nélkül maradó határainak száma adja az algoritmus törlési hibáinak számát, a fölöslegesen behúzott (azaz pár nélkül maradt) gépi határok száma pedig a beszúrási hibák számát.

Az algoritmus tesztelését az OASIS-Numbers adatbázis (http://www.inf.u-szeged.hu/oasis) egy részén végeztük el, amely jó minőségben rögzített számbemondásokat tartalmaz. Azt találtuk, hogy 30 ezredmásodperces hibaküszöb mellett a 10488 valódi szegmenshatárra 28262 beszúrási és 88 törlési hiba jutott. Ez azt jelenti, hogy az algoritmus kb. négyszer annyi határt húz be, mint kellene, viszont a kihagyások száma egy százalék alatt van! A négyszeres „határtúladagolás" soknak tűnhet, de valójában ez csak töredéke annak, ahány lehetőséget a felismerő végigvizsgál, így a szegmentáló használata látványos gyorsulást eredményezett a felismerő futásában. Meg kell jegyeznünk azonban, hogy az OASIS-Numbers adatbázis csak számbemondásokat tartalmaz, amelyeket egyrészt elég gondos artikuláció jellemez, másrészt rengeteg hangkapcsolat egyáltalán nem fordul elő bennük. Ezért az algoritmus további tesztelését tervezzük az MTBA adatbázison.

3 A kézi szegmentálás elvei és tapasztalatai

Az MTBA adatbázis kézi szegmentálásának elkészülte után igyekeztünk összegyűjteni a tapasztalatainkat. Egyik szempontunk az volt, hogy mennyiben sikerült betartani az előzetesen rögzített szegmentálási alapelveket. A másik nézőpontunk az volt, hogy a szegmentálók hogyan kezelték az ejtésvariációkat, azaz mihez kezdtek olyankor, ha a gép által javasolt fonetikai átirat eltért az általuk érzékelttől. Ezekből fontos tanulságok vonhatók le az automatikus átirat, illetve a felismerők szótárának összeállítására nézve, így főleg az ilyen jellegű tapasztalatainkat összegezzük az alábbiakban.

3.1 A szegmentálás szempontjai

A beszédhangok a lineáris hangszerkezetben hangkapcsolatokat képeznek. A hangok gyakran nem különíthetők el élesen egymástól, rövidebbek-hosszabbak köztük az átmenetek, ezért a munka során azt tapasztalhatjuk, hogy a beszédhanghatárt nem lehet mindig egyértelműen bejelölni. Gyakran a spektrogram és a hullámforma összehasonlítása, de még a meghallgatás sem ad objektív fogódzót a határ pontos bejelölésére. Ezekre az esetekre az MTBA adatbázis feldolgozása elkezdése előtt az alábbi elveket dolgoztuk ki [6]:

- A szegmentumhatárokat érdemes a nullátmenetekhez igazítani.
- Zöngés hangok esetében ez a pozitív nullátmenetet jelenti. Nagyon precízen kell jelölni a határt. Zöngétlen hangoknál 1 ms pontossággal jelölhető a hang kezdete.
- A magánhangzó kezdetét a zönge indulásánál kell jelölni (zöngétlen hang után).
- A zárhangok, affrikáták kezdetét a megelőző hang utolsó periódusa előtt jelöljük.
- A magánhangzó-magánhangzó vagy magánhangzó-rezonáns mássalhangzó kapcsolatokban a határt az átmeneti rész 50%-ánál jelöljük be. Ilyenkor némileg pontatlan a bejelölés, mert a hangok kettéválasztása bizonytalan [7].
- Előfordulhat, hogy egy-egy hangot többszöri visszahallgatás után sem lehetett azonosítani. Ennek jelzésére a [cut] kódot lehet használni.

3.2 Szegmentálási tapasztalatok

A kézi szegmentálást végzők az előző pontban felsorolt elvek közül az egyiktől rendszeresen eltértek, anélkül, hogy ezt észrevették volna. Ez az elv a következő volt: "A magánhangzó kezdetét a zönge indulásánál kell jelölni (zöngétlen hang után)". Ez az angolban vagy németben jól alkalmazható elv a magyarban csak a réshangok utáni magánhangzók kezdetének a kijelölésére használható. Az angolban vagy a németben zárhangok esetén is jól működik, mert ezekben a nyelvekben a zönge jóval a zár felpattanása után kezdődik.[8] A magyarban viszont a felpattanáskor szinte azonnal rezegni kezdenek a hangszalagok. Ha a szegmentálók valóban a zönge indulásánál húzták volna be a vonalat, akkor a zárhangból nem sok maradt volna. Szerencsére nem ezt tették, hanem a fülükre hallgatva a felpattanó zörejt meghagyták a zárhangnak.

A szegmentálási szempontok közül az, hogy a zárhangok, affrikáták kezdetét a megelőző hang utolsó periódusa előtt jelöljük, újdonság a korábbi gyakorlathoz képest. Korábban ezt a szakaszt az egyszerűség kedvéért a megelőző hanghoz soroltuk. A határvonal balra tolását az indokolja, hogy így a fel nem pattanó zárhangok felismerése is lehetővé válik [4]. Ezeknek az aránya ugyan nem jelentős a magyarban, de annyi azért van belőlük, hogy ne mondjunk le róluk. Leggyakrabban azonos képzési helyű orrhangok előtt maradhat el a zár felpattanása, pl. *népmese, kötni,* vagy hasonulásos alakokban: *pillanatnyi, vadnyúl,* de néha megnyilatkozás végén is megfigyelhető.

A beszédfelismerési folyamatnak abban a szakaszában, amelynek a végén eljutunk a beszédhangokhoz, általában még nincs szükség szótárra. Tisztán statisztikai alapon

[8] A felpattanás és a zöngeindulás közötti zöngétlen szakaszt nevezik hehezetnek.

is hozzá lehet rendelni hangszakasztípusokhoz beszédhang-szimbólumokat. De már itt is vannak nem egyértelmű esetek.

A szótagvégi *l* gyakran eltűnik, és megnyújtja az előtte levő magánhangzót. A megnyúlt magánhangzó nem azonos a rövid magánhangzó hosszú párjával: minőségében ugyanolyan marad, mint a rövid magánhangzó, csak az időtartama lesz hosszabb, pl. *elment* > *ēment* [E:mEnt], és nem *ément* [e:mEnt] [3]. Ilyen esetekben a szegmentáló hiába keresi az *l*-et, mert az nincs ott. Jó beszédfelismerést valószínűleg az eredményezhet, ha a szegmentáló nem próbálja önkényesen rövid [E]-re és [l]-re bontani a hosszú [E:]-t, hanem a valósághoz hűen csak egy szegmentumot vesz fel, és ezt [E:]-vel jelöli. Ezt a beszédfelismerő program olyan bemeneti adatként kezeli, amelynek az egyik lehetséges kimenete az /E/+/l/ fonémakapcsolat (illetve <el> betűkapcsolat). Azért csak az egyik lehetséges kimenete, mert fizikailag ez az [E:] nem különíthető el az egymást szünet nélkül követő két [E]-től, pl. *leesett.*[9] A szegmentálók ugyan mindig két szegmentumként elemzik a *leesett*-nek az *l* és *s* közötti szakaszát, de a beszédfelismerő program nem mindig kap kézzel szegmentált anyagot. Kézi szegmentálás híján viszont pusztán a hullámformából lehetetlen eldöntenie, hogy az elemzendő hangszakasz az /El/ vagy az /EE/ fonémakapcsolat megvalósulása-e. Ez csak szótár segítségével lehetséges. Ha a szótárban benne van, vagy a szótár elemeiből kialakítható az, hogy *leesett,* de az nem, hogy *lelsett,* akkor az [E:] ~ [EE] bemeneti adat /E]/+/E/ kapcsolatként interpretálandó.

A szegmentáló embernek megvan az az előnye a géppel szemben, hogy ismeri a nyelvet, és előre el tudja dönteni a többféleképp is interpretálható hangszakaszokról, hogy melyik a helyes interpretáció. Valójában nem is szoktunk tudatában lenni annak, hogy amit elemzünk, azt másképp is lehetne elemezni. A *kiutazik* [k]-ja és [t]-je közötti szakaszt mindenki két szegmentumra bontja ([i]-re és [u]-ra), a *kijut* [k]-ja és [t]-je közötti szakaszt pedig háromra ([i]-re, [j]-re és [u]-ra). De csak akkor, ha az egész szót hallják. Ha kivágjuk a *kiutazik* „iu"-ként értelmezett szakaszát és a *kijut* „iju"-ját, és a szegmentálóknak csak a kivágott szakaszt adjuk oda, rögtön nem lesz egyértelmű, hogy melyik szakasz hány szegmentumból áll. Ennek az az oka, hogy az [i]-t szünet nélkül követő magánhangzók olyankor is [j]-vel kapcsolódnak az [i]-hez, ha azt a helyesírás nem jelöli, pl. *fi*[j]*am, Pistá*[j]*ék.* A jó helyesíró szegmentálók, ahol nem írnak <j>-t, ott nem is keresnek [j]-t. Az automata szegmentáló viszont csak úgy mehet biztosra, ha a kétféleképp is értelmezhető hangszakaszokról szótár segítségével dönti el, hogy melyik értelmezést fogadja el.

A hasonult és összeolvadt alakok visszaalakítása már a beszédhangok megállapításával záruló szakasz után következik, amelyben a szótár még fontosabb szerepet kap. Azt, hogy a *pénzt* szóban a [t] előtti zöngétlen beszédhang egy zöngés fonémának a megvalósulása, és emiatt nem <sz>-szel írandó, hanem <z>-vel, vagy hogy az *aludj, hagyj* és *higgy* hosszú [d':]-je csak kiejtésben azonos, leírva mind a három más, csak szótár segítségével állapítható meg. Ez a szótár természetesen többféleképp is létrehozható: begépeltethető emberekkel is, de a gép is kialakíthat magának egy olyan, az emberi szótáraktól esetleg jelentősen eltérő szótárszerű képződményt, amelynek alapján megtippelheti, hogy egy beszédhanghoz mikor milyen betű vagy betűkapcsolat rendelhető hozzá a legvalószínűbben. A szótárban való keresés hatékonyságát növelheti, ha beszédfelismerő program kellő „tapasztalatokkal" rendelkezik arról, hogy

[9]Budapesti köznyelvi adat. A kétféle *e*-t ismerő nyelvváltozatokban a *leesett* két *e*-je nem egyforma

melyik beszédhangot milyen betű jelölheti. Ezek a tapasztalatok hosszadalmas betanítással nyelvész közreműködése nélkül is megszerezhetők, de egy jól algoritmizálható hangtani leírás valószínűleg jelentősen csökkentheti a betanítási időt.

A programtervező matematikusok és nyelvészek együttműködése nem mindig egyirányú. Bár az esetek többségében a programtervezők építik be saját rendszereikbe a nyelvészet által felhalmozott ismereteket, időnként vissza is fordul ez a folyamat. Az /l/ fonémának egy olyan allofónja, amelyről a magyar nyelvű szakirodalomban nem írnak, egy beszédfelismerő program betanítása során jelentkezett olyan gyakorisággal, hogy feltűnt a szegmentálóknak. Arról, hogy a *-tlan/-tlen* képzőben nagyon gyakran zöngétlen az /l/, [5]-ben olvashatunk először.

Bibliográfia

[1] Kiss J.: *Magyar dialektológia.* Budapest, Osiris Kiadó, 2001

[2] Mihajlik, P. és Tatai, P.: Automatikus fonetikus átírás magyar nyelvű beszédfelismeréshez, Beszédkutatás 2001.

[3] Nádasdy Á., Siptár P.: A magánhangzók, in. Kiefer F. (szerk.): *Strukturális magyar nyelvtan* 2: *Fonológia.* Budapest: Akadémiai Kiadó, 42–182.

[4] Sejtes Gy., Zsigri Gy.: Hangátmenetek a beszédfelismerésben, in: Alexin Z., Csendes D. (eds.): Magyar Számítógépes nyelvészeti Konferencia 2003, Szeged, pp. 176–181.

[5] Tóth, L., Kocsor, A.: Az MTBA magyar telefonbeszéd-adatbázis kézi feldolgozásának tapasztalatai, Beszédkutatás 2003, pp. 134-146.

[6] Vicsi, K., Tóth, L., Kocsor, A., Gordos, G., Csirik, J. MTBA-Magyar nyelvű telefonbeszéd-adatbázis, Híradástechnika, LVII. 2002/8, Budapest, pp. 35-43.

[7] Vicsi K., Vig A.: Az első magyarnyelvű beszédadatbázis, Beszédkutatás '98, MTA Nyelvtudományi Intézete, Budapest 1998, pp. 163-177

[8] Vicsi, K., Szaszák, Gy.: A magyar nyelv kiejtésvariációi és felhasználásuk a beszédfelismerésben II., Beszédkutatás 2003, pp. 163-176

[9] Zsigri, Gy., Kocsor, A., Tóth, L., and Sejtes, Gy.: Phonetic Level Annotation and Segmentation of Hungarian Speech Databases, accepted for Acta Cybernetica

English papers and abstracts

LiLe project: Database as 'dynamic corpus'

Zoltán Bódis, Judit Kleiber, Éva Szilágyi, Anita Viszket

Department of Linguistics, University of Pécs

Our research team has been engaged in developing a linguistic lexicon in the form of an MS-SQL database.

The technology provides the opportunity to store lexical units (morphemes) as well as applying underlying representations, and also using the well-known features of unification morphology; in a dynamically expandable structure. In our system not only the phonological, morphological, syntactic and semantic features of a morpheme are stored as records, but also the rules operating within or between the lexical items. Consequently, the set of rules can be dynamically expanded as well. The theory behind the definition-method of the rules is GASG, a totally lexicalist grammar by Gábor Alberti.

The structure of rules for describing the certain language is determinated by the describers, and the grammars are linked through a semantic representation. So in describing lexical elements only the semantic features are common (universal), other features are freely formed; in this way we can provide useable lexicon for any grammatical model.

By using our database, we can build up a corpus which is called 'dynamic', because it doesn't contain existing (ever existed) wordforms, but deducted elements and rules, so the possible words, expressions or even sentences of the given state of language can be generated – consequently, it models our competency.

This supports several purposes, like developing computational linguistic applications at our department, and we also wish to support teaching Hungarian language in public or higher education or as a foreign language with our lexicon as a teaching device.

Due to the structure of the system, our program not only decides between correct and incorrect morpheme strings, but it is able to name the rules used as the basis of the decision, and that is a useful help in langage teaching or developing language consciousness; besides, non-native speakers lacking competency can be supplied or helped out. Even, the generating algorhythm can operate on selected set of elements of the database (morphemes or rules) and this gives an opportunity to demonstrate or practice certain phenomenon of the language in an interesting way, for example, by switching rules on and off.

The current version of the program is developed in an object-oriented environment (Delphi), because we have found this to be the easiest way of building user-friendly interfaces but we plan to develop web-based surfaces as well, applying MS-SQL's built-in procedures which make possible to retrieve data in XML-format (which is generally used for data-storing in corpus linguistics).

Software Package for Supporting Information Extraction Research

Zoltán Alexin[1], Tibor Gyimóthy[1], János Csirik[1]

University of Szeged, Department of Informatics,
Árpád tér 2., Szeged, Hungary
e-mail:{alexin,gyimothy,csirik}@inf.u-szeged.hu

Keywords: Information Extraction, Natural Language Processing, Shallow Syntactic Parsing

The research of the technology for IE (Information Extraction) is a dynamically emerging field of NLP (Natural Language Processing). Collecting the relevant information by computers from the vast amount of texts appearing on the Internet and providing it in brief form is a daily need in politics, economy, science, and even in intelligence services. While IR (Information Retrieval), which is one of the characteristic features of web browsers, aims to present the needed documents in original form to the users, the task of IE includes marking and then collecting the relevant information from the texts as well. Hence text compression and IE are closely related.

IE systems do not intend to fully understand or analyze the documents in detail. The main requirements set against them are big capacity, speed, and an acceptable accuracy. They are usually satisfied with identification of major actors in the sentences without complete syntactic parsing. To accomplish this they do shallow parsing. In the identification of actors, *named entities* play important role. Recognizing person names, companies, geographic locations, cities frequently cited in newspapers is done in a separate processing step based on a large lexicon independently from morphological parsing.

In this paper a software package is presented, which is developed at the Department of Informatics at the University of Szeged for supporting IE research. The most important design concept of it was modularity, so that individual components can be developed independently. Modules can be run separately or in a batch. The output of each module can be followed up and be checked. These features are important at the beginning of the research phase, when different experimental approaches are tried. The standardized communication between two subsequent modules eases changing one module to another and so selecting the best one for a specific task.

The presented system was applied for processing short business news. The MTI-Eco, Business+ service [1] has been used to create a database of 6453 articles for training and testing purposes. Most development efforts are directed to modules being in key positions of the processing, such as POS-taggers, shallow syntactic parsers and semantic pattern (semantic-frame) recognizers.

[1] Hungarian News Agency (Magyar Távírati Iroda), `http://www.mti.hu`

Semantic frame matching, and the automatic evaluation of an Information Extraction system

Richard Farkas[1], Kinga Konczer[2], György Szarvas[1]

[1] Hungarian Academy of Sciences, University of Szeged;
Research Group on Artificial Intelligence,
6720 Szeged, Aradi vertanuk tere 1., Hungary,
{rfarkas, szarvas}@inf.u-szeged.hu
[2] University of Szeged, Hungary
kinga.konczer@hungary.org

Abstract: The Frametagger is a semantic pattern matching software designed to identify the actors in short business news, developed by the Human Language Technology Group of the University of Szeged. The module is based on the semantic frames and tables developed in the NKFP 2/017/2001 project by the Research Institute for Linguistics, and extended by us later on, and is the final module of the Information Extraction Toolchain developed in Szeged. In this paper, we introduce a Benchmark algorithm as well, which is made to give a realistic insight how the development, or replacement of each module in the toolchain effects the results/accuracy of the whole Information Extraction System.

Semantic frame matching and evaluation

The goal of Information extraction is collecting and marking relevant information in documents. Systems in practice usually concentrate on the identification of the relevant actors in texts (in spite of a more general semantic role labeling task, where the goal is to identify the syntactic structure for each verb), without doing much detailed syntactic or semantic analysis.

In our system we applied shallow parsing and a semantic frame set for identifying relevant actors. The frames describe events by giving syntactic and semantic constraints of sentence constituents that play relevant roles in that event. In our case a pattern matching is done on parsed texts and each frame's target word and other roles.

The benchmark algorithm we developed compares the results of the system to a some gold standard files containing the hand-made annotation of semantic roles concerning the same frame set the IE toolchain uses. We give not only an accuracy relative to manual annotation, but also try to guess (in case of differences between the gold standard and the toolchain results) which module's errors misled the whole system to find other, or no fit at all. By now the toolchain works with 70.25% accuracy according to the benchmark, with the majority of errors caused by different NP structure annotation, or matching constituent that play indifferent roles but are ambiguous to a role in the frame we use.

A New Approach to Automatic Term Extraction

Ádám Kis
SZAK Publishers Ltd.
adam.kis@szak.hu

Balázs Kis
MorphoLogic Ltd.
kis@morphologic.hu

Gábor Pohl
Pázmány Péter Catholic University,
Fac. of Information Technology
pohl@itk.ppke.hu

An ideal term extraction system is capable of finding terms in previously unknown source texts without human intervention, and with large recall and precision. However, term occurrences have semantic, syntactic and discourse-related characteristics, so this task raises a modelling problem common to all fields of computational linguistics: for the sake of efficiency and feasibility, most linguistic phenomena must be assigned one or more surface characteristics.

This paper starts with emphasizing some definition problems related to term extraction. Then the authors describe a project aiming at the development of a term extraction system using a new approach. The model they employ here traces back the problem of term recognition to two basic attributes of lexemes: their terminological position and terminological role. The paper addresses the representation of nets of terms, and calls attention to the language- and topic-dependent nature of terminology.

The authors conclude the paper by comparing the present approach to those described in literature, and possible evaluation procedures.

GeLexi project: Machine Translation based on Total Lexicalism

Gábor Alberti, Judit Kleiber, Anita Viszket

Linguistics Dept., University of Pécs

As last year, the bacic aim of our research team is to verify that computational linguistics is worth returning to the pure theoretical (generative) linguistic basis. Our crucial argument still relies on a double (parallel computational and linguistic) chance: to use a significantly greater number of huge patterns than earlier due to the immense increase in memory capacity; and to work out a formal grammar, showing the distribution of capacity advantageous in modern computer science: "minimal processing - maximal database". This latter chance has something to do with the sweeping lexicalist turn in generative linguistics.

This year we focus on the demonstration of a new (totally lexicalist) approach to machine translation which is based on the two-way application of our parser (accepting SL sentences, generating TL sentences). At the moment we can translate English sentences into Hungarian and vice versa, on a small corpus.

Total lexicalism means that every kind of information is stored in the lexicon, and the only syntactic "weapon" is *unification*, which means that there is no need for generating phrase structure trees. In this grammar lexical items are not (fully inflected) words but morphemes (stems and affixes), which is relevant on the syntactic and semantic "level" as well.

The input of our parser is a string (a sentence), and there are several outputs: the list of the relevant lexical items, the list of the established syntactic relations, a discourse-semantic representation (based on ReALIS, which is a developed version of Kamp's DRT), and a copredicative network, which is a level between syntax and semantics.

There is an isomorphic relation between the semantic representation of the Hungarian and the English versions of a sentence, which suggests the idea of applying our semantic parser in the area of machine-aided translation. It would take only a few hours to teach an intelligent, say, English-speaking person how to interpret a semantic representation containing English names of predicators; whilst it would take years to teach her even a basic level of, say, Hungarian.

To provide "real" machine translation, we can generate sentences by using the equivalents of the relevant lexical items of the source language, and presuming variables at specific template positions (for case and agreement marking affixes). Only grammatical sentences can be generated, because all the candidates go through the same parsing mechanism as they were SL sentences.

An important innovation of this approach is that there is no need for different mechanisms to translate from and into different languages. The only task is to elaborate grammars of more and more languages to achieve translation, because the frame we propose is universal.

Hunglish: a statistical Hungarian–English Machine Translation system

Péter Halácsy*, András Kornai**, László Németh*, András Rung*, István Szakadát*, Viktor Trón***, Dániel Varga*

This paper lays out our plans for a simple Hungarian to English machine translation system and describes our accomplishments so far. In the preparatory stage we collect a parallel corpus (so far, we have collected about a quarter of the planned 100m words) and use this to verify a pre-existing, but in many ways overly broad, Hungarian–English dictionary.

In the simplest version of the planned system, we use the resulting dictionary to create a lattice of translation possibilities, and find the best path through the lattice by Viterbi search. Since Hungarian word order is kept in the translation, the result is not expected to be fully grammatical, let alone idiomatic, English, though it is already expected to be serviceable for cross-language information retrieval, where word order is typically ignored.

More complex versions of the system, which analyze the source in greater detail (NP-level chunks and predicate-argument structure) and transfer this analysis to the target, will be built incrementally on top of the simpler system. As with other NLP projects at the Budapest Institute of Technology Media Research and Education Center, all dictionaries, corpora, and software created in the project will be made available under a non-restrictive (LGPL) open source license.

* Budapesti Műszaki Egyetem Média Oktató és Kutató Központ, {hp, nemeth, runga, szakadat, daniel}@mokk.bme.hu

** MetaCarta Inc., andras@kornai.com

*** International Graduate College, Saarland University and University of Edinburgh, v.tron@ed.ac.uk

MemoQ: A New Approach to Computer-assisted Translation

István Lengyel, Balázs Kis, Gábor Ugray

Kilgray Translation Technologies
{istvan.lengyel;balazs.kis;gabor.ugray}@kilgray.com

Recent surveys and papers on the use of CAT tools among translators (Drugan 2004, Fulford-Granell-Zafra 2004, Somers 2003) have pointed out that only a few translators adopt a complete computer-aided workflow. However, considering translators as a community governed by similar rules and practices and implementing a holistic approach to the translation process yields additional savings which can only be exploited by means of an integrated translation scheme. Kilgray's ambitious MemoQ project – basically an intelligent, language-aware translation memory – will be the first practical implementation of the philosophy discussed here.

The translation market is interested in efficiency, and computer-aided translation is an efficiency tool. Nevertheless, most systems were developed in a modular way, and in the first stage of development, all of them attempted at satisfying the needs of an individual translator working in isolation, in an off-line environment. Later, the Internet challenged translators and changed the market conditions entirely. Translation memories expanded their operations to the Internet, but did not reflect the change in paradigm.

Translation is a project-based activity, and the workflow in a very early stage defines (1) the people working on the project, (2) the resources used within the project.

Translation companies hire project managers to provide for workflow-based efficiency. However, TMs do not take into account the workflow roles. The authors, together with Actiwise Consulting, have developed Forditas.net, a web and e-mail-based workflow automation system. This system – which spans the workflow – can be regarded the vertical network component, because it connects people playing different roles: the coordinator to the translator, the translator to the proofreader, and so on. However, the more we automate this process, from quotations to delivery and invoicing, the more efficient the system will get because of (1) the one-stop web storage of all project resources, preventing data loss, (2) the elimination of time spent on forwarding (waiting time is even more significant), (3) thee partial elimination of file names and full elimination of directory structures, resulting in less confusion.

The horizontal network component, on the other hand, connects users of the same role. Resource sharing and instant messaging contributes to consistency.

Fuzzy algorithms can also be complemented with language-sensitive parsing, providing a drastical increase in efficiency. In morphologically rich languages such as Hungarian, Spanish or Arabic, a language-sensitive operation as basic as word stemming can produce much better results. If we can even parse sentences, and create sentence skeletons (Kis-Gröbler-Hodász, 2004), grammar patterns can also be checked for. However, using intelligent parsing ruins the concept of a translation segment.

An intelligent translation memory, complemented with a multi-dimensional domain system, raises questions. How to keep the interface intuitive if we have at least three results which helps in assessing the quality of match: a fuzzy index, a domain match index, and a grammar index? The translator is only interested in one thing: the best order. Establishing a single composite index is a challenging task, and gives ground to further research.

Learning and recognizing full syntax of sentence

András Hócza

University of Szeged, Department of Artificial Intelligence
6720 Szeged, Árpád tér 2.
hocza@inf.u-szeged.hu
http://www.inf.u-szeged.hu

Keywords: full syntax, machine learning, rule based methods

Full syntax recognition is the process of determining whether sequences of words can be grouped together as noun, adjective, verb, etc. phrases. This information is essential in machine understanding of a sentence from natural language. This means that each word of a sentence and possible word groups must be identified. Phrases often contain another phrases, therefore the phrase structure of a sentence is a tree. An additional feature of syntax a tree that it is coherent, fully cover the sentence and it has only one root which traditionally labeled by S.

Hungarian is an agglutinated language with a rich morphology and relatively free word order, whose properties add difficulties to the full analysis of the Hungarian language compared to Indo-European languages. These difficulties mean that the automatic syntax recognition of Hungarian language is too complicated to solve using experts' rules only. An efficient solution for this problem might be the application of machine learning methods, but it requires a large number of training and test examples of annotated sentences. Since the Szeged Corpus[10] became available, new methods have begun to be developed for syntactically parsing Hungarian sentences. The corpus contains texts from five different topic areas and is currently comprised of about 1.2 million word entries, 145 thousand different word forms, and an additional 225 thousand punctuation marks.

After the completion of the annotation work the Szeged Corpus was then used for training and testing machine learning algorithms to retrieve syntax recognition rules. This paper introduces an application of the RGLearn algorithm that was used to learn syntax tree patterns described by regular expressions. The tree patterns are completed with probability values using error statistics. The syntax parser uses this grammar to build up the best syntax trees of a sentence by backtracking. The results look fairly promising after comparing them to related works. This method was developed as a part of a system which extracts information from short business news texts written in the Hungarian language.

[10] The different versions of the Szeged Corpus are available at http://www.inf.u-szeged.hu/hlt.

Statistical Named Entity recognition for Hungarian

Richard Farkas[1], György Szarvas[1]

[1] Hungarian Academy of Sciences, University of Szeged;
Research Group on Artificial Intelligence,
6720 Szeged, Aradi vértanuk tere 1., Hungary,
{rfarkas, szarvas}@inf.u-szeged.hu

Abstract: In this paper, we present decision tree based statistical Named Entity recognizer system for Hungarian. The model was trained and tested on a segment of the Szeged Corpus, containing short business news articles, collected from MTI (Hungarian News Agency, www.mti.hu). We applied C4.5 for classificaton, and examined the accuracy of the system using training sets of different sizes. For this task we used only numerically encodable information (we excluded the word form itself), which contained some orthographical rules specific to Hungarian, but we trained for the recognition of foreign language proper nouns appearing frequently in business news as well. During the experiments the best results showed an accuracy of 89.6% F measure.

The feature set we used:

- Part of Speech code (for the particular word and for its +/- 4 words context)
- Case code
- Type of the word's first letter (for the particular word and for its +/- 4 words context)
- Contains digit (inside the word form)
- Contains capital letter (inside the word form)
- Contains hyphen (inside the word form)
- Is the word the beginning of a sentence
- Is the word between quotation marks is the sentence
- Word length
- Is the word Arabic or a Roman number
- The quotient of lower case frequency and "ignore case" frequency from Szószablya [4] term frequency dictionary for Hungarian
- The quotient of mid-sentence upper case frequency and all uppercase frequency from Szószablya
- Does one of our dictionaries contain the word (we used dictionaries containing city names, country names, surnames, company types, geographical name endings, stop words (used frequently in lower case inside NEs)) (for the particular word and for its +/- 4 words context)

Open source morphological analyzer

Németh László*, Halácsy Péter* Kornai András**, Trón Viktor***

The HunTools natural language processing toolkit emerged from the SzolSzablya morphological analyzer project at the the Budapest Institute of Technology Media Education and Research Center. In this paper we concentrate on the architecture of the MorphBase morphological component which supports spellchecking, stemming, morphological analysis, and generation in a set of language-neutral routines, and describe the Hungarian-specific resources. Both the Hungarian-specific and the language-neutral parts of the system are available under the open source LGPL license.

The core engine of MorphBase is an extension of the well-known open-source ispell spellchecker, which identifies correctly spelled words by first stripping affixes according to a rule-system and then looking up the stems from a lexicon. Both the rule-system and the lexicon are specified as input files and compiled off-line. Our improved version is similarly language independent (and compatible with ispell file formats) but has significant additional functionality. First, we enabled the output of stripped forms, thereby creating a stemmer. Next, we enabled alternative analyses, both in stemming and providing a full morphological analysis. Finally, we replaced the simple the simple one-pass affix stripping mechanism of ispell by a recursive system in which affixes can be stripped in as many layers as needed. This results in considerable simplification of the lexical resource, as well as increased linguistic transparency and maintainability.

* Budapest University of Technology Centre for Media Research and Education, {nemeth,halacsy}@mokk.bme.hu

** MetaCarta Inc., andras@kornai.com

*** International Graduate College, Saarland University and University of Edinburgh, v.tron@ed.ac.uk

A formalism for encoding morphological annotation in HunTools

András Kornai*, Péter Rebrus**, Péter Vajda* Péter Halácsy***, András Rung**, Viktor Trón†

The MorphBase library contains all word-oriented routines of the HunTools NLP toolkit currently under development at the Budapest Institute of Technology Media Education and Research Center (see `http://www.szoszablya.hu`). Morphological generation requires paradigmatic information as input, e.g. that we wish to generate the accusative form of a noun or the 3sg past form of a verb, and morphological analysis will provide paradigmatic (inflectional) and possibly deeper (derivational) output. There are some standards for encoding morphological structure, such as MSD, but we have not found any that met our requirements for consistency, informativeness, and maintainability.

The paper motivates and describes in detail the encoding used in MorphBase. The treatment of derivational affixes is standard. The encoding of the inflection affixes is based on arranging paradigmatic dimensions in a simple tree structure, where only positive (marked) nodes can branch. This way, markedness conventions can trivially supply a full feature matrix, so that encoding of the unmarked (typically the most frequent) cases can be kept short. Defective paradigms are described by subtrees of the main tree, and a number of "minor categories" such as pronouns are treated quite naturally as being defective instances of their superordinate categories.

* MetaCarta Inc., e-mail: `andras@kornai.com`
** MTA Nyelvtudományi Intézet, e-mail: `{rebrus,vajda}@nytud.hu`
*** BME Média Oktató és Kutató Központ `{hp,runga}@mokk.bme.hu`
† IGK, Saarland University, University of Edinburgh `v.tron@ed.ac.uk`

HunLex - a framework for morphological dictionaries

Viktor Trón*

In this article, we present HunLex, a morphological resource-specification framework and resource compiler tool which is being developed as part of the Budapest Institute of Technology Media Education and Research Center's HunTools NLP toolkit (see `http://www.szoszablya.hu`).

HunLex offers a formalism for specifying a base lexicon and morphological rules which can then serve as a central database capable of providing language-specific knowledge to a variety of NLP tools. The prototype implemented for the Szószablya project at the BIT is able to provide optimized lexical resources for the HunTools MorphBase routines (spell-checker, stemmer, morphological analyzer/generator). These resources are compiled offline from the central lexicon and grammar in a highly configurable way so that users can fine-tune these resources according to their needs.

The motivation behind HunLex came from two opposing types of requirements lexical resources are supposed to fulfill: (i) scalability, maintainability, extensibility; and (ii) optimized format for the application. The constraints in (i) favour one central, redundancy-free, abstract, but transparent specification, while (ii) requires various application-specific, and potentially redundant, optimized formats. In order to reconcile these requirements, HunLex introduces an offline layer which mediates between the two levels of resources: a central database conforming to (i), which is ideal for human maintanance, and the various specific formats that are inputs to software modules conforming to (ii) for performance. HunLex is used to compile the base resources into an application-specific format (called dic and aff files in the case of the MorphBase routines) in a configurable way. This includes the choice of format for algorithm (spell-checking, stemming, morphological analysis or generation), selection of morphemes, grouping of morphemes to be stripped as a cluster (with one rule application), selection of morphophonological features that are to be observed or ignored, depth of recursive rule application, selection of registers and degree of normativity based on usage qualifiers in the database.

The HunLex framework is used in the development of an open-source morphological database (lexicon and grammar) for the Hungarian language in a collaboration between the Research Institute for Linguistics and the MERC Lab, which aspires to be the most complete and accurate account of Hungarian morphology published so far.

* International Graduate College, Saarland University and University of Edinburgh, `v.tron@ed.ac.uk`

The first morphological analyzer for Nganasan

Attila Novák

MorphoLogic Ltd. 1126 Budapest Orbánhegyi út 5.,
novak@morphologic.hu

This article presents a morphological analyzer for *Nganasan*, a small language belonging to the Northern Samoyed branch of the Finno-Ugrian language family. Creating this analyzer is part of a project the aim of which is to create annotated corpora and other electronically available linguistic resources for a number of small members of the Uralic language family. The project was initiated by Various Hungarian research groups specialized in Finno-Ugric linguistics and a Hungarian language technology company, MorphoLogic.[1]

Ngansan turned out to be especially interesting among the languages involved in the project. On the one hand, it is a language on the verge of extinction (the number of native speakers is below 500 by now, most of them are middle-aged or old), so its documentation is an urgent scientific task. On the other hand, its morphology and especially its phonology is so complex that the implementation of the analyzer turned out to be a real challenge. Using the formalism of the morphological analyzer engine called *Humor*, which we successfully applied to other languages involved in the project, turned out not to be feasible in the case of Nganasan. Finally, we used the regular relation calculus based toolset of Xerox Corp. (namely, *xfst*) to create the analyzer.

In Nganasan, a quite morphology-independent surface phonology plays an important role in shaping the form of words. The very productive gradation processes are governed by a complicated set of constraints on surface syllable structure. Gradation is a systematic alternation of obstruents in syllable onsets governed in the case of Nganasan by various factors such as vowel length and the presence of a coda in the preceding syllable, the presence of a coda in the current syllable, and whether the syllable is in an odd or even position within the word. The syllabification of certain segments or clusters is exceptional and there are also apparent lexical exceptions to the general gradation patterns.

The Humor formalism uses an 'item-and-arrangement' model of morphology where feature-based allomorph adjacency restrictions are the primary device for constraining word structure. Gradation in Nganasan is difficult to formalize as a set of allomorph adjacency restrictions because the segments involved in determining the outcome of the process may belong to non-adjacent morphemes. Moreover, gradation is just a small part of the complicated system of dozens of interacting productive and lexicalized morphophonological and phonological alternations. The Xerox finite-state calculus (specifically *xfst*), which fortunately became freely available and easily accessible for non-commercial purposes in 2003, proved to be more easily applicable to this language.

[1] Complex Uralic Lingustic Database, NKFP 5/135/2001.

Evaluation of school time by content analysis of arguing compositions of 8th grade students

Zsuzsanna Huszár[1], Dr. András Sramó[2]

[1]PTE BTK Teacher Training Institute, 7624 Pécs, Ifjúság útja. 6.
huszped@tki.pte.hu,
[2]PTE KTK Department of Business Informatics, 7622 Pécs, Rákóczi út 80.
sramo@igyfk.pte.hu

Keywords: temporal structures, aspectuality, cronotopology, context analysis, visualization, Galois-graph

Abstract. This article is a direct continuation of our previous study (Exploration of temporal structures by qualitative and quantitative text analysis), which was prepared for last year's conference; the study examines and summarizes the content analysis performed on the compositions written by 8th grade students which was presented in 2003. In addition to the classification system used previously, we introduce a binary coding of text attributes as the basis for manual computational analysis. Based on the manual input the study suggests other possible mechanical representations and points out the function these new representations have in our research.

Multilingual Database of Proverbs

Hrisztova-Gotthardt Hrisztalina

No Institute Given

PhD Student
University of Pécs, Faculty of Humanities, Doctoral School Linguistics
E-mail: xpucu@freeweb.hu

Keywords: proverbs, proverb corpora, multilingual online database, XML, language world view

Within the confines of my thesis for the doctor's degree, proposing the contrastive analysis of Bulgarian, Hungarian and German Proverbs in the light of world view, arose the problem of the absence of usable corpora for my research. Thus I made a resolution to elaborate and create a proverb database with multilingual support.

This article introduces the prospective multilingual proverb database. The Internet database based on a web-like schema presents the characteristics of the proverbs and the pertaining information in detail - the origin, source, style, classification and formal characteristics among others. The database allows of creation of semantical connections between the proverbs in different languages and connects the partial or full linguistic equalities.

The public access to the database, the possibility of detailed search, extension and modification might support the work of the linguists and folklorists, publishers of textbooks, and the work of translators and teachers who could use it as a corpora for their scientific, editorial and translation work.

The Hyper-Morpheme

Language Technology and Text

Ádám Kis
SZAK Publishers Ltd.
adam.kis@szak.hu

Balázs Kis
MorphoLogic Ltd.
kis@morphologic.hu

This paper addresses the representation of the entirety or specific parts and levels of text, from the aspect of their computational treatment. The approach differs from the 'traditional' syntactic or analysis-based aspects of computational linguistics. Instead, it looks at the contents of the text as a complex entity. The authors use three application examples – references (or links), searching and machine translation – to show that the (modelling) problems of computational linguistics all have one common root: namely, the limitations of the computers' pattern matching capabilities. The principal question here is, is one able to overcome these limitations, or is it possible to compare or search for texts based on their information content? This paper raises tasks for researchers without having a solution ready for them.

LAS VERTICUM: 'TIME' MODULE

Bea Ehmann

Institute for Psychological Research of the Hungarian Academy of Sciences
1132 Budapest, Victor Hugo u. 18-22.
ehmannb@mtapi.hu

Abstract. The paper demonstrates the time modules and categories of the LAS VERTICUM supralexical content analysis software pack, as well as the related psychological validation studies. Content analytic results on subjective time experience were compared to El Meligi's psychometric constructs in construct validation, and to Antonovsky's Sense of Coherence factors in criterion validation. Several significant correlations were found.

LAS VERTICUM: 'Characters and Functions' Module

Bernadette Péley

PÉCS UNIVERSITY, Institute of Psychology,
7624 Pécs, Ifjúság útja 6.
peley@btk.pte.hu

Abstract. The paper presents the „Characters and Functions" module of the LAS VERTICUM software. Codes for characters are mother, father, parents, family, and relatives. Functions involve 21 categories, such as helper, enemy, neglecting person, abandoning person, threatening person, etc. The module automatically codes all characters in the subject's life story and all psychological functions that belong to each character. The paper also presents results of the validity studies performed to prove the psychological relevance of the coding categories.

Autobiographical Narrative Perspective and Emotion Regulation

Tibor Pólya

Institute for Psychological Research of the Hungarian Academy of Sciences
1132 Budapest, Victor Hugo u. 18-22.
polya@mtapi.hu

Abstract The paper defines the concept and formal markers of autobiographical narrative perspective (retrospective, reexperiencing and experiencing forms), as well as linguistic markers which may identify particular narrative forms in autobiographical narrative. A module for automatic identification of autobiographical narrative perspective forms, developed in cooperation with Morphologic Ltd, is demonstrated, and so are the results of the reliability tests of the module. Finally, main results of our study for the verification of emotion regulatory function of autobiographical narrative perspective are described.

Speechreading

László Czap

University of Miskolc, Department of Automation
3515 Miskolc, Egyetemváros
czap@mazsola.iit.uni-miskolc.hu

Summary

Automatic speechreading systems through their use of visual information to support the acoustic signal have been shown to yield better recognition performance than purely acoustic systems, especially when background noise is present. In this paper an answer is sought to the most important questions of speechreading: Which features can represent visual information well? How can they be extracted? An intelligibility study was carried out to see which parts of the face give the most support to speechreading. The whole face, mouth or lips were visible dubbed with noisy voice. Visual support to speech perception of the image ellipse model is compared to that of the parts of the natural face.

It is generally agreed that most visual information is carried by the lips. The inner lips are especially important and a remarkable improvement comes from the visibility of teeth and tongue. Geometric features and the intensity factor of the oral cavity are discussed as a means of visual speech representation. Much of the research in speechreading systems is focused on the crucial problem of feature extraction. How can it best transform a sequence of images into feature values that facilitate recognition? The process should be fast, robust, and yield as much information as possible carried by the fewest number of features, removing redundant and linguistically irrelevant information. Whereas there is no one favorite way of representing visual speech there are impressive methods that all require tracking the inner and outer contours of the lips. A novel feature extraction method based on a similarity study is proposed that does not need tracking of the lips.

Efficiency of the geometric and pixel based features are compared on a continuous speech recognition task. Pixel based features can represent the visual speech better than the geometric ones.

Semi-syllables and diphones were compared as candidates for basic linguistic elements of automatic speech recognition for an agglutinating language. The diphone based recognition highly outperformed the semi syllable one on the audio-visual and the acoustic database as well. The conclusion is that context sensitive elements can yield better performance.

Speech recognizer model-building experiments at the level of acoustics and phonetics, on behalf of developing a speech recognizer for medical reporting

Szabolcs Velkei, Klára Vicsi

Budapest University of Technology and Economics,
Department of Telecommunications and Media Informatics,
Laboratory of Speech Acoustics
1117 Budapest, Magyar tudósok krt. 2.

E-mail: vicsi@tmit.bme.hu, velkei@tmit.bme.hu

In this article a HMM based speech recognition system is introduced, which has been developed at the Laboratory of Speech Acoustics. The final aim is the development of a middle sized continuous speech recognizer. New methods have been developed in the acoustical preprocessing, in the statistical model-building, moreover phonetically, phonological and morphemic levels have been involved for the recognition process. In the first year the optimization of the acoustic, phonetic level was prepared.

The developed systems, called MKBP 0.8 were compared with the well known HMM toolkit, HTK. Different evaluation methods were examined and explained, how the results depend on the methods of the evaluation.

The comparison research shows that by the optimization of the acoustic preprocessing and the development of the acoustics-phonetics models we can increase the recognition accuracy and decrease the time of the processing. Of course, the involvement of the higher linguistic levels will increase significantly the recognition accuracy, but with a better starting it is possible to obtain better ending results.

Hungarian speech database for computer-using environment in offices

Klára Vicsi[1], András Kocsor [2], Csaba Teleki[1], László Tóth[2]

[1] BME Távközlési és Médiainformatikai Tanszék,
Beszédakusztikai Kutatólaboratórium, Sztoczek u. 2.,
1111 Budapest, Magyarország
{vicsi, teleki}@tmit.bme.hu
http://alpha.tmit.bme.hu/speech
[2] MTA-SZTE Mesterséges Intelligencia Kutatócsoport,
Aradi vértanuk tere 1.,
6720 Szeged, Magyarország
{kocsor, tothl}@inf.u-szeged.hu

Speech databases were recorded in different offices, laboratories, and homes. Recordings in all scenes were prepared by using two parallel synchronized recording systems. One of the recording systems is the so-called reference system, where a close talking microphone (Monacor EMC 100) and a good quality sound card (Hercules Muse Pocket USB 5.1) and a laptop (Gericom Webshox) were used. The second recording system was the so-called varied system, where different microphones, sound cards and PCs were applied.

Description of the database:

- *The recording form was 16 bits and 16 kHz*
- *332 speakers;*
- *Twelve sentences and twelve words per speaker from a phonetically balanced text, composed in accordance with special Hungarian phonetic expectances;*
- *Large variations of different microphones, sound cards and PCs were used;*
- *Computer-using environment in offices, homes and laboratories;*
- *The whole material of database is annotated and one third (100 speakers) is segmented and labeled by hand.*

Automatic segmentation of continuous speech at word- and phrase level by using suprasegmental parameters

Klára Vicsi, György Szaszák, Gábor Borostyán

Budapest University of Technology and Economics,
Department of Telecommunications and Media Informatics,
Laboratory of Speech Acoustics
{vicsi, szaszak}@tmit.bme.hu http://alpha.tmit.bme.hu/speech/

In our article we are searching for the answer of a question, whether it is possible to segment the continuous speech at the boundaries of words and phrases by examination of the change of fundamental frequency and energy level in time. We want to increase the robustness of the speech recognizers at linguistic level by the detection of boundaries of words and phrases. In this way we can significantly decrease the searching space during decoding.

In Hungarian language if stress is present, it marks always the first syllable of the word stressed. Thus if we can detect these stressed syllables, than we can detect the boundaries of the word. We describe the developed different searching algorithms.

For the evaluation of these algorithms we carried out examinations the BABEL Hungarian speech database. The results were the best, when the algorithm used the time series of the fundamental frequencies and the energies together. Perhaps the accuracy of the decisions by using these algorithms will decrease in spontaneous speech compared with the ones demonstrated here, but these results show that it is worth to continue our work on this field.

Névmutató